Variational Optimization of finite Projected Entangled Pair States

Dissertation
zur Erlangung des Doktorgrades
der Naturwissenschaften
(Dr. rer. nat.)

dem

Fachbereich Physik
der Philipps-Universität Marburg

vorgelegt von

Markus Scheb
aus
Dinslaken

Marburg, 2021

Vom Fachbereich Physik
der Philipps-Universität
als Dissertation angenommen am: 26.03.2021

Erstgutachter: Prof. Dr. Reinhard M. Noack

Zweitgutachter: Prof. Dr. Peter Lenz

Tag der mündlichen Prüfung: 29.03.2021

Hochschulkennziffer: 1180

Zusammenfassung

Die zugrundeliegende Dissertation ist im Bereich der Vielteilchentheorie der Quantenmechanik angesiedelt. Im Speziellen wird das Hubbard Modell betrachtet, welches seit den 1960er Jahren als Testumgebung für stark korrelierte Elektronensysteme dient und trotz mehrerer Jahrzehnte intensiver Forschung immer noch nicht vollständig beschrieben ist. Von besonderem Interesse ist dabei eine starke, repulsive Elektron-Elektron Wechselwirkung und eine Dotierung, die etwas unterhalb halber Füllung liegt. Die Abstoßung zwischen Elektronen begünstigt eine antiferromagnetische Ordnung, während die Anwesenheit von Löchern zu einer frustrierten Konfiguration führt, die nur bedingt mithilfe störungstheoretischer Methoden berechnet werden kann. Grund für das Interesse an diesem Punkt im Phasendiagramm ist die Vermutung, dass es sich um eine vereinfachte Abbildung der Cuprat Supraleiter handelt, deren Paarungsmechanismus trotz ihrer Entdeckung im Jahr 1986 immer noch nicht vollständig erfasst ist.

Zahlreiche analytische und numerische Methoden wurden entwickelt, um die Grundzustandsenergie dieses Parametersatzes und anderer anspruchsvoller Modelle zu berechnen. Die Methode dieser Arbeit fällt in das Gebiet der Tensor Netzwerke, welche als eine Art Datenkompression quantenmechanischer Wellenfunktionen aufgefasst werden können. Der prominenteste und am weitesten fortgeschrittene Algorithmus dieses Gebietes ist die Dichtematrix - Renormierungsgruppe (DMRG), welche eine extrem zuverlässige Methode zur Grundzustandsberechnung eindimensionaler Quantensysteme ist. In diesem Kontext wird hier ein Prototyp für die Verallgemeinerung der DMRG auf zwei Dimensionen vorgestellt. Dafür wird die Darstellung der Wellenfunktion als Projected Entangled Pair State ("Projizierter verschränkter Paar Zustand"), oder einfach PEPS, genutzt, dessen quantenmechanische Verschränkung auf die Struktur eines zweidimensionalen Gitters maßgeschneidert ist. Eine variationelle Optimierung erfolgt dann lokal und skaliert linear mit der Systemgröße.

Die Arbeit ist wie folgt gegliedert: Kapitel 1 ordnet das physikalische Problem und die numerische Methode in den historischen Kontext ein. Als nächstes trifft Kapitel 2 mehrere Vorbereitungen. Zuerst wird die iterative Diagonalisierung skizziert (Kapitel 2.1), die an sich bereits ein Algorithmus zur Berechnung extremaler Eigenwerte ist und den Kern der variationellen Optimierung darstellt. Es folgt eine ausführliche Beschreibung der Symmetrien des Hubbard Modells (Kapitel 2.2), da deren Ausnutzung essentiell für die effiziente Implementierung von Tensor Netzwerken ist. Schließlich wird das Wigner Eckart Theorem hergeleitet (Kapitel 2.3), das für nicht-Abelsche Symmetrien benötigt wird.

Kapitel 3 geht auf die quantenmechanische Verschränkung ein und wie diese innerhalb der Vielteilchentheorie ausgenutzt werden kann. Kapitel 3.1 präsentiert das AKLT

Modell, welches als Motivation zur Tensor-Netzwerk-Darstellung von Grundzuständen dient. Anschließend wird die Von-Neumann Entropie erklärt (Kapitel 3.2), die die Verschränkung innerhalb einer Wellenfunktion quantifiziert. Kapitel 3.3 nimmt Bezug auf physikalische Systeme, indem die Skalierung der Entropie mehrerer Modelle aufgezählt wird.

In Kapitel 4 werden die elementaren Bausteine eines Tensor-Kalküls erläutert, das sowohl Abelsche als auch nicht-Abelsche Symmetrien berücksichtigt. Die Betonung liegt dabei weniger auf mathematischer Vollständigkeit oder Präzision, als auf Verständlichkeit und Pragmatismus. Oft werden Mechanismen nur für Beispiele erklärt, mit der Unterstellung, das der allgemeine Fall selbsterklärend ist. Zuerst werden Tensoren im Allgemeinen definiert (Kapitel 4.1), wie die Symmetrien auszunutzen sind und wie sie am besten abzuspeichern sind. Es folgt eine Erläuterung der Permutation der Indizes (Kapitel 4.2), der paarweisen Kontraktion zweier Tensoren (Kapitel 4.3), der Zusammenführung und Trennung von Indizes (Kapitel 4.4) und der Faktorisierung eines Tensors in zwei (Kapitel 4.5). Zum Schluss wird ein effizientes Verfahren zur Kontraktion mehrerer Tensor präsentiert (Kapitel 4.6), was in der Regel den Flaschenhals von Tensor-Netzwerk Algorithmen darstellt.

Kapitel 5 als ganzes liefert eine kompakte Erklärung der DMRG in der Sprache der Matrix-Produkt Zustände. Obwohl die DMRG selbst nicht Ziel dieser Forschungsarbeit ist, lohnt es sich aus rein didaktischer Sicht, diese zuerst im Prinzip zu verstehen, bevor man sich mit PEPS beschäftigt. Mehrere Konzepte können dann als Sprungbrett zu zwei Dimensionen genommen werden.

Kapitel 6.1 geht endlich auf PEPS selbst und deren Definition ein. Da nur offene Randbedingungen betrachtet werden, müsste man genauer von fPEPS (finite PEPS) reden, in Abgrenzung zu iPEPS (infinite PEPS), welche Translationsinvarianz berücksichtigen (siehe Titel der Arbeit). Als nächstes wird ein Schema präsentiert, das die Darstellung eines lokalen Hamiltonians auf die Topologie des PEPSs anpasst (Kapitel 6.2). Anschließend wird im Detail erläutert, wie Erwartungswerte innerhalb des PEPS - Formalismus approximativ zu berechnen sind (Kapitel 6.3), was eine zentrale Schwierigkeit darstellt. Kapitel 6.4 fügt schließlich alle Bausteine dieser Arbeit zusammen und definiert die variationelle Optimierung eines PEPS zur Grundzustandsberechnung zweidimensionaler Quantensysteme.

In Kapitel 7 wird der vorgestellte Algorithmus auf das zweidimensionale Hubbard Modell angewandt. Dabei wird zuerst der Einfluss der approximativen Berechnung von Erwartungswerten auf die variationelle Optimierung untersucht und der Fehler quantifiziert. Zum Schluss werden ein paar Testsimulationen für ein 3x3 und ein 8x8 Gitter durchgeführt. Der Algorithmus liefert eine stabile Konvergenz der Energie und lokale Ladungs- und Spindichten, die qualitativ mit denen aus vorhergehenden Publikationen

übereinstimmen. Allerdings ist er in seiner aktuellen Version aufgrund einiger technischer Subtilitäten noch nicht in der Lage, Grundzustandsenergien bis auf mehrere signifikante Stellen zu reproduzieren. Kapitel 8 geht schließlich im Detail auf den Entwicklungsstatus der Optimierung ein, welche Verbesserungen noch zu implementieren sind und welche physikalischen Phänomene im Anschluss analysierbar wären.

Contents

List of Figures

List of Tables

1 Introduction

The history of investigating strongly correlated electron systems can be traced back to studies of metal-insulator transitions by Nevill Mott, in particular, his paper on transition metals in 1949 [1]. While it was, up to this point, believed that the single-particle properties of a material determine whether it is a metal or an insulator, Mott predicted that, at least for nickel oxide, the repulsion between electrons could lead to an insulating gap. As he noted in his later review [2], his theory was met with scepticism until the studies of narrow energy bands in the 1960s [3–7] were able to substantiate his prediction. Hubbard [4], most notably, considered the Hamiltonian of interacting electrons in the Wannier basis:

$$H = \sum_{i,j,\sigma} t_{i,j}\, c^\dagger_{i,\sigma}\, c_{j,\sigma} + \sum_{i,j,k,l,\sigma_1,\sigma_2} V_{i,j,k,l}\, c^\dagger_{i,\sigma_1}\, c^\dagger_{j,\sigma_2}\, c_{k,\sigma_2}\, c_{l,\sigma_1}\,. \tag{1.1}$$

Here i, j, k, and l are indices of atomic sites, σ, σ_1, and σ_2 represent spin configurations, $c^\dagger_{i,\sigma}$ and $c_{i,\sigma}$ are fermionic creation and annihilation operators of Wannier states on site i, $t_{i,j}$ are the tunneling amplitudes, and $V_{i,j,k,l}$ are the matrix elements of the Coulomb interaction. Even if one is provided with reasonable approximations for $t_{i,j}$ and $V_{i,j,k,l}$, for instance, by using density functional theory [8, 9], solving Eq. (1.1) is usually not feasible due to the complexity of $V_{i,j,k,l}$. Hubbard's essential idea was to relate narrow energy bands to large interatomic spacing. The Wannier functions can then be approximated by atomic s-functions with a small radius, which leads to a single dominant interaction term U for two electrons with antiparallel spin on the same site:

$$H = \sum_{i,j,\sigma} t_{i,j}\, c^\dagger_{i,\sigma}\, c_{j,\sigma} + U \sum_i c^\dagger_{i,\uparrow}\, c^\dagger_{i,\downarrow}\, c_{i,\downarrow}\, c_{i,\uparrow}\,. \tag{1.2}$$

Due to the screening of the electron gas, interaction terms of electrons on different sites decay exponentially with distance and can be neglected. Additionally, the $t_{i,j}$ are often reduced to nearest-neighbor hopping only. The Hamiltonian in Eq. (1.2) has since been known as the *Hubbard model*, and is able to explain the aforementioned metal-insulator transition, also known as *Mott transition*. Gutzwiller conducted similar research within that time [3, 5], with a special focus on the magnetic ordering of ground states. He made the same approximation as Hubbard and introduced a variational parameter that takes the interaction into account. For an almost empty band, the ground state turns out be non-magnetic, while it exhibits magnetic order for an almost full band. Kanamori [5] and Nagaoka [7] also studied ferromagnetism in narrow bands, the latter with respect to face-centered cubic and hexagonal lattices.

In the subsequent 20 years, the Hubbard model was investigated with the primary focus

on observables at low temperatures and on the Mott transition. Lieb and Wu [10] used the Bethe ansatz, which was originally devised for spin-chains [11], and adapted it to the one-dimensional Hubbard model at half filling. This led to the Lieb-Wu-equations, which yield a conductor for $U = 0$ and an insulator for $U > 0$. Their approach was justified by Shastry [12] in 1986, who proved the integrability of the Hubbard model in one dimension. Brinkman and Rice [13], on the other hand, examined Eq. (1.2) using the Gutzwiller ansatz and found a Mott transition at finite U, which is, at least in one dimension, at odds with Lieb and Wu's result. Ovchinnikov [14] expanded Hubbard's investigation by determining low-energy excitations above the ground state in one dimension. He verified Lieb and Wu's analytic conclusion that there is no Mott transition by showing that the difference between the ground-state energy and the energy of the first excited state is infinitely small in the thermodynamic limit. In a very short letter, Langer and Mattis [15] gave upper and lower bounds for the ground state energy. Shiba [16], building upon the work of Lieb and Wu, calculated the magnetic susceptibility as a function of the electron density for various onsite repulsions and found a steady increase in all cases. Beni et al. [17] considered Hubbard chains in the strong-coupling limit for finite temperatures and calculated the partition function, the magnetic susceptibility, and the electron mobility. Takahashi [18] subsequently gave a rigorous derivation of the specific heat. He found a discontinuity at $U = 0$ for half filling and a continuous behaviour away from half filling. Emery [19] turned his attention to the strong-coupling limit for both attractive and repulsive interaction. For negative U, the model becomes equivalent to a Bose gas, which can be related to the formation of Cooper pairs within the BCS theory of superconductivity [20]. For positive U and half filling, the Hubbard model is equivalent to the Heisenberg model. After the introduction of the renormalization group (RG) by Wilson [21], Hirsch [22] used a real-space RG approach to study the Hubbard model at half filling in multiple dimensions and was able to verifiy the results of both Lieb and Wu [10] and Brinkman and Rice [13]: In one dimension, the ground state is insulating for all $U > 0$, while two and three dimensions give rise to a Mott transition at a finite U. Five years later, Hirsch [23] additionally found an effective attraction between electrons for a strong onsite repulsion, using Monte-Carlo simulations.

With the discovery of high-temperature superconductors in 1986 by Bednorz and Müller [24], the focus of interest shifted to a new class of materials, the cuprate superconductors. Shortly thereafter, Anderson [25] provided a qualitative explanation for the underlying pairing mechanism with his theory of resonating valence bonds (RVBs), which he originally invented for the ground state of spins on a triangular lattice [26]. In this mechanism, electrons on neighboring sites are connected by a singlet state (i.e., a valence bond), which can be moved due to higher order corrections, hence, they resonate. In the half-filled Hubbard model with strong onsite repulsion, the ground state is simply an insulating

antiferromagnet. In the nearly half-filled Hubbard model, however, the antiferromagnetic pairing remains, while the presence of a few holes allows for an electric current and leads to an RVB state. Due to the finite energy within the valence bonds, the RVB state cannot be easily broken up by impurities or phonons, which yields a superconducting phase. Note how this stands in stark contrast to BCS superconductivity [20], in which the pairing is normally mediated by lattice oscillations.

Superconductivity has thus become an additional subject of investigation within the Hubbard model. Gros [27] elaborated on the competition between the paramagnetic RVB state and the antiferromagnetic phase. Scalapino [28] investigated the pairing mechanism for near-nested Fermi surfaces and found a tendency towards d-wave superconductivity, in which the interaction between sites has an alternating sign for increasing distance. In particular, two electrons repel each other on the same site, attract each other if they are nearest neighbors, again repel each other for next-nearest neighbors, and so on. Hirsch and Lin used both Monte Carlo simulations [29] and exact diagonalization [30] to study the repulsive, two-dimensional Hubbard model at fillings between 0.25 and 0.75, and found no superconductivity. At the same time, Bickers et al. [31] calculated two particle correlations with diagrammatic techniques and proposed a phase diagram as a function of temperature and doping. Their results yield a magnetic phase for average densities between 1 and approximately 0.96, and d-wave superconductivity for low temperatures and densities between 0.96 and 0.85, which contradicts the previous result. Subsequently, van Dongen and Vollhardt [32] solved the Hubbard model analytically for infinite-range hopping, where $t_{i,j} = t$ for all i and j, and found the ground state to be insulating at half-filling for $U > 0$ and conducting otherwise. Lieb examined the Hubbard model from a mathematical perspective and proved two general theorems [33], which do not depend on the dimension or the boundary conditions: The first is a condition for to the total spin of the ground state, which is unique in all cases. The second proves the possibility of itinerant ferromagnetism. Metzner and Vollhardt [34] considered the case of infinite dimensions, which can be solved using diagrammatic techniques. The weak-coupling limit is then taken as an approximation for three dimensions. Georges and Kotliar [35] elaborated on this idea and showed how the mean-field picture of the Hubbard model becomes more accurate for increasing dimension. For infinite dimensions, the Hubbard model is exactly described by the mean field approach and is equivalent to the single impurity Anderson model. White et al. [36] calculated the ground state of the Hubbard model in two dimensions numerically, for both zero and finite temperature. In accordance with previous results, they found long-range antiferromagnetic order at half filling, its absence at quarter-filling, and a tendency towards d-wave pairing near half filling.

In the early 1990s, White introduced the density-matrix renormalization group (DMRG)

[37, 38], a variational algorithm for the calculation of the ground-state properties of one-dimensional quantum systems. While the costs of exact diagonalization grow exponentially with system size, the DMRG is able to numerically calculate the ground state of any local, one-dimensional Hamiltonian accurately. Soon after, Östlund and Rommer [39] investigated the convergence of the DMRG in the thermodynamic limit. They found that the resulting wave function could be expressed as a matrix product state (MPS), which had its precursors in the AKLT-states by Affleck et al. [40], and the finitely correlated states by Fannes et al. [41]. MPSs have since become a popular tool for the calculation of one-dimensional systems, both within and outside the framework of the DMRG. While MPSs can, in principle, be applied to two-dimensional systems by mapping a lattice onto a chain, the computational effort scales exponentially with the width of the system, as physically local correlations need to be mapped onto virtual, long-range correlations [42].

It was soon realized that MPSs are a subclass of tensor network states (TNSs), all of which utilize quantum mechanical entanglement to represent wave functions. For reasons which will be explained thoroughly in this thesis, TNSs are promising candidates for the representation of ground states in not just one, but also in two dimensions, and are at the forefront of exploring strongly correlated quantum systems, where the Hartree-Fock approximation, the random phase approximation, exact diagonalization, density functional theory, renormalization groups and Quantum Monte Carlo algorithms all reach their computational limits. One prominent type of TNSs are tree tensor network states (TTNSs), which were proposed by Shi et al. [43]. Due to their acyclic topology, the most sensible field of application for TTNSs is the study of molecules in quantum chemistry, which was examined in Refs. [44–47]. Slightly modified, trees can be used to map multiple physical sites onto one supersite, which is equivalent to the block spin renormalization group by Kadanoff [48]. This leads to the multiscale entanglement renormalization ansatz, which was introduced by Corboz et al. [49, 50] in 2009, but has not been used extensively in recent years.

The main topic of this thesis, however, are projected entangled pair states (PEPSs), which were invented by Verstraete et al. in 2004 [51, 52] as the natural generalization of MPSs to two dimensions. It was soon shown that PEPSs represent ground states of local, two-dimensional quantum systems [53–55] and that they can be used to classify quantum phases through their isometric form [56]. Murg et al. conducted one of the first numerical simulations and provided benchmark results for hard-core bosons [57] and frustrated spin-systems [58]. Schuch et al. connected PEPSs to our focus of interest, high-temperature superconductivity, by using them to study RVB states in the Kagome lattice. The effort to investigate fermions using PEPSs has since been led by Corboz, who has specialized in the infinite PEPS (iPEPS) algorithm, originally introduced by Jordan [59]. In this method, translational invariance is emulated by replicating a unit cell infinitely in both

spatial dimensions. In a series of papers [60–63], Corboz et al. used both ordinary PEPS and iPEPS to examine the t-J-model, a simplified version of the Hubbard model. They found a characteristic stripe structure, in which both the charge and the spin density of electrons oscillate over multiple sites and compete with d-wave pairing.

The last major milestone relevant to this thesis was set just recently in 2017, when Corboz and others joined forces to study the two-dimensional Hubbard model for strong coupling and near half-filling [64], which is where Anderson predicted superconductivity [25]. They compared the numerical results of iPEPS, the DMRG, a Quantum Monte Carlo algorithm, and density matrix embedding theory and found stripe structures consistent with previous theoretical research. While all methods yielded the same qualitative behaviour of the charge and spin densities, they still differed in detail and were not entirely consistent with measurements performed on real materials. This shows that the quest of investigating strongly correlated electron systems, especially in two dimensions and in the thermodynamic limit, is far from over.

In this context, the present thesis introduces a prototype for the variational optimization of finite projected entangled pair states for open boundary conditions. First, the iterative diagonalization is outlined (Sec. 2.1), which is used to determine extremal eigenvalues. It is followed by a detailed description of symmetries within the Hubbard model (Sec. 2.2), since their exploitation is essential for an efficient implementation of tensor networks. Afterwards, the Wigner-Eckart theorem is derived, which is needed for non-abelian symmetries.

Chapter 3 concerns itself with quantum mechanical entanglement and how it can be utilized in many body physics. Sec. 3.1 presents the AKLT model, which serves as a motivation for tensor network representations of ground states. Subsequently, the von Neumann entropy is elucidated (Sec. 3.2), which quantifies the entanglement inside of wavefunctions. Sec. 3.3 makes a connection to physical systems by describing several models and their scaling of the entropy.

Chapter 4 explains elementary tensor operations that take both abelian and non-abelian symmetries into account. The emphasis is less on mathematical rigor than on intelligibility and pragmatism. Mechanisms are often explained using examples, assuming the general case is self-explaining. First, tensors are defined in general (Sec. 4.1), in particular, how their symmetries are taken advantage of and how to store them. There follows an explanation of the permutation of indices (Sec. 4.2), the pairwise contraction of tensors (Sec. 4.3), the fusion and splitting of indices (Sec. 4.4), and the factorization of a tensor into two (Sec. 4.5). Finally, we present an efficient method for contracting multiple tensors, which usually poses the main bottleneck in tensor-network algorithms.

Chapter 5 delivers a compact explanation of the DMRG in the language of matrix

product states. Although the DMRG itself is not the goal of this research project, it is worthwhile to describe its general principles, before moving on to PEPS. Multiple concepts can then be used as a stepping stone to treating two dimensions.

Sec. 6.1 finally takes on PEPS itself. Since only open boundary conditions are considered, we have to consider finite PEPS (fPEPS), as opposed to iPEPS, which is fundamentally based on translational invariance. Subsequently, a scheme that adapts the representation of a local Hamiltonian to the topology of a PEPS is presented (Sec. 6.2). This is followed by a detailed explanation of how to determine expectation values approximately (Sec. 6.3), which is one of the central difficulties of the algorithm. Sec. 6.4 finally puts all of the pieces together to define the overarching algorithm for the variational optimization of fPEPSs as used to determine ground states of two-dimensional quantum systems.

In Chapter 7, the fPEPS algorithm is applied to the two-dimensional Hubbard model. First, the influence of the approximate calculation of expectation values is investigated and the error is quantified. Afterwards, a few test simulations are conducted on 3x3 and 8x8 lattices. The algorithm yields a stable convergence of the energy and local charge and spin densities. The local observables resemble those of previous publications qualitatively. However, our version of fPEPS is not yet able to reproduce ground state energies up to more than a couple of significant figures due to some technical subtleties. Finally, Chapter 8 discusses the development status of the optimization in detail, what improvement are pending, and what physical phenomena could be analyzed in the future.

2 Preliminaries

2.1 Iterative Diagonalization

Consider the eigenvalue problem of a symmetric $n \times n$ - Hamiltonian

$$H \left|\varphi_i\right\rangle = E_i \left|\varphi_i\right\rangle \,, \tag{2.1}$$

with n being the dimension of the Hilbert space. In order to calculate the full spectrum, one has to apply the Householder transformation to bring H into tridiagonal form, followed by the QL algorithm. However, in the field of strongly correlated electrons, one is usually only interested in the low-energy spectrum, or, more specifically, the ground state $\left|\varphi_0\right\rangle$, which is the eigenstate with the lowest energy eigenvalue. Performing a full diagonalization would be wasteful in this case, which is why we use algorithms based on the power method. In this approach, we start with an inital vector $\left|\nu_0\right\rangle$ and apply H multiple times

$$\left|\nu_n\right\rangle = H^n \left|\nu_0\right\rangle = H^n \sum_i \left|\varphi_i\right\rangle \left\langle\varphi_i|\nu_0\right\rangle = \sum_i E_i^n \left\langle\varphi_i|\nu_0\right\rangle \left|\varphi_i\right\rangle \,. \tag{2.2}$$

For each successive application of H, the $\left|\varphi_i\right\rangle$ whose E_i have a higher absolute value become increasingly dominant, which is why the space spanned by $\{\left|\nu_0\right\rangle, \left|\nu_1\right\rangle, ..., \left|\nu_n\right\rangle\}$, called Krylov-subspace, describes the proximity of extremal eigenstates. The ground state can, in principle, be determined by $\lim\limits_{n\to\infty} \left|\nu_n\right\rangle$, given that $\left|\nu_0\right\rangle$ has at least some overlap with it.

The Lanczos algorithm [65] refines this idea. Instead of building the space by mere application of H, we additionally orthogonalize against the existing basis. In particular, we start with an initial vector $\left|\nu_0\right\rangle$ just as before, but calculate the next one via

$$\left|\nu_1\right\rangle = H \left|\nu_0\right\rangle - a_0 \left|\nu_0\right\rangle \,, \quad a_0 = \frac{\left\langle\nu_0\right| H \left|\nu_0\right\rangle}{\left\langle\nu_0|\nu_0\right\rangle} \,, \tag{2.3}$$

and all others using

$$\left|\nu_{n+1}\right\rangle = H \left|\nu_n\right\rangle - a_n \left|\nu_n\right\rangle - b_n^2 \left|\nu_{n-1}\right\rangle \,, \quad a_n = \frac{\left\langle\nu_n\right| H \left|\nu_n\right\rangle}{\left\langle\nu_n|\nu_n\right\rangle}, \quad b_n^2 = \frac{\left\langle\nu_n|\nu_n\right\rangle}{\left\langle\nu_{n-1}|\nu_{n-1}\right\rangle} \,. \tag{2.4}$$

We now verify the orthogonality of the $\left|\nu_n\right\rangle$ and determine the matrix elements $\left\langle\nu_i\right| H \left|\nu_j\right\rangle$ through overlaps. First, we calculate

$$\left\langle\nu_n|\nu_{n+1}\right\rangle = \left\langle\nu_n\right| H \left|\nu_n\right\rangle - a_n \left\langle\nu_n|\nu_n\right\rangle - b_n^2 \left\langle\nu_n|\nu_{n-1}\right\rangle = -b_n^2 \left\langle\nu_n|\nu_{n-1}\right\rangle \,. \tag{2.5}$$

Since $\left\langle\nu_0|\nu_1\right\rangle = 0$, recursive application of Eq. (2.5) gives

$$\left\langle\nu_n|\nu_{n-1}\right\rangle = 0 \,. \tag{2.6}$$

The norm of a Lanczos vector ,

$$\langle \nu_n | \nu_n \rangle = \langle \nu_n | H | \nu_{n-1} \rangle - b_{n-1}^2 \langle \nu_n | \nu_{n-2} \rangle \ , \tag{2.7}$$

combined with

$$\langle \nu_{n-1} | \nu_{n+1} \rangle = \langle \nu_{n-1} | H | \nu_n \rangle - \langle \nu_n | \nu_n \rangle \tag{2.8}$$

yields

$$\langle \nu_{n-1} | \nu_{n+1} \rangle = b_{n-1}^2 \langle \nu_n | \nu_{n-2} \rangle \ . \tag{2.9}$$

As $\langle \nu_2 | \nu_0 \rangle = 0$, we obtain

$$\langle \nu_n | \nu_{n-2} \rangle = 0 \tag{2.10}$$

using recursion and thus

$$\langle \nu_n | \nu_n \rangle = \langle \nu_n | H | \nu_{n-1} \rangle \equiv c_n \, . \tag{2.11}$$

By induction, it can be proven that $\langle \nu_i | \nu_j \rangle = 0$ and $\langle \nu_i | H | \nu_j \rangle = 0$ for $i < j + 1$, which yields a tridiagonal structure.

The Krylov subspace is expanded until a threshold, such as $\langle \nu_{n+1} | \nu_{n+1} \rangle < \epsilon$, is reached and the effective Hamiltonian has the form

$$T = \begin{pmatrix} a_0 & c_1 & 0 & & \\ c_1 & a_1 & c_2 & 0 & \\ 0 & c_2 & a_2 & \ddots & 0 \\ & 0 & \ddots & \ddots & c_n \\ & & 0 & c_n & a_n \end{pmatrix} \ . \tag{2.12}$$

Here T is usually taken to be smaller than H in its full basis, which gives a dramatic speedup. One proceeds by diagonalizing T using the QL algorithm, which gives eigenvectors in the basis $|\nu_i\rangle$:

$$|\varphi_j\rangle = \sum_i c_i |\nu_i\rangle \ . \tag{2.13}$$

If $|n\rangle$ are the original basis vectors, the extremal eigenvectors can be constructed as

$$|\varphi_j\rangle = \sum_{i,n} c_i \langle n | \nu_i \rangle |n\rangle \ . \tag{2.14}$$

Note that the $|\nu_i\rangle$ still must be normalized.

A further improvement was proposed by Davidson [66]. Suppose we have a Krylov subspace spanned by $\{|\nu_0\rangle, ..., |\nu_n\rangle\}$, an effective Hamiltonian $H_{\text{eff}} = \langle\nu_i| H |\nu_j\rangle$, the approximate ground state within that space $|\tilde{\varphi}_n\rangle$, and the according energy $\lambda_n = \langle\tilde{\varphi}_n |H| \tilde{\varphi}_n\rangle$. The initial condition for $n = 0$ reads $|\tilde{\varphi}_0\rangle = |\nu_0\rangle$ and $H_{\text{eff}} = \lambda_0 = \langle\nu_0 |H| \nu_0\rangle$. The Ritz variational principle states that λ_n is always an upper bound for the actual ground state energy E_0, towards which we want to move. The next iteration is then determined by the correction to the ground state $|\varphi_0\rangle$, defined by

$$|z\rangle = |\varphi_0\rangle - |\tilde{\varphi}_n\rangle \ , \tag{2.15}$$

which yields

$$(H - E_0\mathbb{1}) |z\rangle = -(H - E_0\mathbb{1}) |\tilde{\varphi}_n\rangle \ . \tag{2.16}$$

Since E_0 is not known, we make a first approximation by using λ_n instead:

$$(H - \lambda_n\mathbb{1}) |z\rangle \approx -(H - \lambda_n\mathbb{1}) |\tilde{\varphi}_n\rangle = -|r_n\rangle \ , \tag{2.17}$$

where $|r_n\rangle$ is the residual vector. Solving Eq. (2.17) for $|z\rangle$ would lead to the best expansion of the Krylov subspace. Unfortunately, it is too expensive in the case of bare exact diagonalization and can be even impossible if it is embedded in a variational tensor network algorithm, since in this case H is fragmented into multiple tensors and is much too large to be calculated as a whole. Instead, Davidson suggested to approximate H by its diagonal elements D, which leads to the inverse iteration

$$|z\rangle = -(D - \lambda_n\mathbb{1})^{-1} |r_n\rangle \ , \tag{2.18}$$

using the preconditioner $(D - \lambda_n\mathbb{1})$. The vector $|z\rangle$ is then orthonormalized against the existing basis $\{|\nu_0\rangle, ..., |\nu_n\rangle\}$ through the Gram-Schmidt process and is added to it as $|\nu_{n+1}\rangle$. Afterwards, the effective Hamiltonian H_{eff} is expanded and the new approximate ground state $|\tilde{\varphi}_{n+1}\rangle$, along with its energy λ_{n+1}, is calculated using full diagonalization. This whole scheme is repeated until either the norm of the residual vector falls below a certain threshold, or until a predetermined maximum number of Davidson steps is reached. The last $|\tilde{\varphi}_n\rangle$ is then picked as the best approximation to the ground state.

The Davidson algorithm can be easily adapted to the generalized eigenvalue problem

$$H |\varphi_i\rangle = E_i N |\varphi_i\rangle \tag{2.19}$$

with a symmetric matrix N. All norms and overlaps are then calculated through $\langle\nu_i| N |\nu_j\rangle$ instead of $\langle\nu_i|\nu_j\rangle$ and the residual vector by $r_n = (H - \lambda_n N) |\tilde{\varphi}_n\rangle$ instead of $r_n =$

$(H - \lambda_n \mathbb{1}) |\tilde{\varphi}_n\rangle$.

For a more thorough explanation of iterative diagonalization, we refer to Refs. [67–69].

2.2 Symmetries in the Hubbard model

In physics, both analytical methods and numerical algorithms benefit enormously from the exploitation of symmetries. In quantum mechanics, in particular, the symmetries of a given Hamiltonian H are determined by the complete set of linearly independent Hermitian operators Q_i that commute with H. The Q_i correspond to observables and can be diagonalized independently, usually in advance. States $|\psi\rangle$ to which H is applied can then be prepared so that they are already eigenstates of all Q_i,

$$Q_i |\psi_{q_1,\ldots,q_n}\rangle = q_i |\psi_{q_1,\ldots,q_n}\rangle \ , \tag{2.20}$$

where the eigenvalues q_i are called *quantum numbers* or *charges*. If we now apply H,

$$H |\psi_{q_1,\ldots,q_n}\rangle = |\varphi_{q_1,\ldots,q_n}\rangle \ , \tag{2.21}$$

the new state φ is still exactly the same eigenstate as ψ to all Q_i as $[H, Q_i] = 0$. The q_i are thus *conserved* and the Hamilton matrix H is block-diagonal with respect to all quantum numbers:

$$\mathrm{H} = \langle \varphi_{q'_1,\ldots,q'_n} |H| \psi_{q_1,\ldots,q_n}\rangle \propto \delta_{q'_1,q_1} \ldots \delta_{q'_n,q_n} \ . \tag{2.22}$$

The computational effort of any algorithm that operates on H can thus be drastically reduced. For example, calculating the full spectrum now amounts to diagonalizing multiple smaller matrices in independent symmetry sectors and omitting all entries that are zero due to conservation laws.

Here we are interested in the application of symmetries to the Hubbard model. In its general form, the Hamiltonian reads

$$H = H_t + H_U + H_\mu + H_B \tag{2.23}$$

$$H_t = \sum_{i,j,\sigma} t_{i,j} \left(c^\dagger_{i,\sigma} c_{j,\sigma} + \mathrm{h.c.} \right) \tag{2.24}$$

$$H_U = U \sum_i c^\dagger_{i,\uparrow} c^\dagger_{i,\downarrow} c_{i,\downarrow} c_{i,\uparrow} \tag{2.25}$$

$$H_\mu = -\mu \sum_{i,\sigma} c^\dagger_{i,\sigma} c_{i,\sigma} \tag{2.26}$$

$$H_B = -B \sum_i \left(c^\dagger_{i,\uparrow} c_{i,\uparrow} - c^\dagger_{i,\downarrow} c_{i,\downarrow} \right) , \tag{2.27}$$

where Eq. (1.2) has been extended by a term that couples the electrons to a chemical potential μ, and thus controls the particle number and a term that couples the spins to a magnetic field B along the z-axis. Here, we do not take into account discrete spatial symmetries due to the deviation from half filling. The usage of conservation of momentum is possible, but, unfortunately, leads to a non-local Hamiltonian. This may not pose an issue for exact-diagonalization methods, but renders tensor network algorithms intractable due to a violation of the area law, as will be discussed in Sec. 3.3.

Most useful for our purposes are continuous, global symmetries, described by Lie groups. One way of elucidating them is to rewrite the kinetic terms as scalar products of spinors [70], for instance,

$$H_t = \sum_{i,j} t_{i,j} \left(\Psi^\dagger_i \Psi_j + \Psi^\dagger_j \Psi_i \right) , \tag{2.28}$$

with

$$\Psi^\dagger_i = \left(c^\dagger_{i,\uparrow}, c^\dagger_{i,\downarrow} \right)^T , \quad \Psi_i = \left(c_{i,\uparrow}, c_{i,\downarrow} \right)^T . \tag{2.29}$$

The two-dimensional structure indicates an SU(2)-symmetry, which, in this case, corresponds to the electronic spin. The associated generators, S^i_+, S^i_-, and S^i_z, can be expressed through fermionic operators as follows:

$$S^i_+ = \Psi^\dagger_i \hat{S}_+ \Psi_i = \begin{pmatrix} c^\dagger_{i,\uparrow} \\ c^\dagger_{i,\downarrow} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} c_{i,\uparrow} \\ c_{i,\downarrow} \end{pmatrix} = c^\dagger_{i,\uparrow} c_{i,\downarrow} , \tag{2.30}$$

$$S^i_- = \Psi^\dagger_i \hat{S}_- \Psi_i = \begin{pmatrix} c^\dagger_{i,\uparrow} \\ c^\dagger_{i,\downarrow} \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} c_{i,\uparrow} \\ c_{i,\downarrow} \end{pmatrix} = c^\dagger_{i,\downarrow} c_{i,\uparrow} , \tag{2.31}$$

$$S^i_z = \Psi^\dagger_i \hat{S}_z \Psi_i = \begin{pmatrix} c^\dagger_{i,\uparrow} \\ c^\dagger_{i,\downarrow} \end{pmatrix} \begin{pmatrix} 1/2 & 0 \\ 0 & -1/2 \end{pmatrix} \begin{pmatrix} c_{i,\uparrow} \\ c_{i,\downarrow} \end{pmatrix} = \frac{1}{2} \left(c^\dagger_{i,\uparrow} c_{i,\uparrow} - c^\dagger_{i,\downarrow} c_{i,\downarrow} \right) . \tag{2.32}$$

The full Hubbard Hamlitonian exhibits an SU(2) symmetry if it commutes with all three total generators,

$$S_+ = \sum_i S^i_+ , \quad S_- = \sum_i S^i_- , \quad S_z = \sum_i S^i_z , \tag{2.33}$$

which is best verified using the following elementary commutation relations:

$$\left[c^\dagger_{i,\tau}\, c_{i,\tau'}, c^\dagger_{j,\sigma}\right] = c^\dagger_{i,\tau}\, \delta_{i,j}\, \delta_{\tau',\sigma} \tag{2.34}$$

$$\left[c^\dagger_{i,\tau}\, c_{i,\tau'}, c_{j,\sigma}\right] = -c_{i,\tau'}\, \delta_{i,j}\, \delta_{\tau,\sigma} \tag{2.35}$$

$$\Rightarrow \left[c^\dagger_{k,\tau}\, c_{k,\tau'}, c^\dagger_{i,\sigma}\, c_{j,\sigma}\right] = c^\dagger_{k,\tau}\, c_{j,\sigma}\, \delta_{k,i}\, \delta_{\tau',\sigma} - c^\dagger_{i,\sigma}\, c_{k,\tau'}\, \delta_{k,j}\, \delta_{\tau,\sigma}\,. \tag{2.36}$$

Roman and Greek literals represent lattice sites and spins, respectively.

We first determine the commutators of the kinetic term:

$$\left[c^\dagger_{k,\tau}\, c_{k,\tau'}, H_t\right] = \sum_{\sigma,i,j} t_{i,j} \left(\left[c^\dagger_{k,\tau}\, c_{k,\tau'}, c^\dagger_{i,\sigma}\, c_{j,\sigma}\right] + \left[c^\dagger_{k,\tau}\, c_{k,\tau'}, c^\dagger_{j,\sigma}\, c_{i,\sigma}\right]\right) \tag{2.37}$$

$$= \sum_{\sigma,i,j} t_{i,j} \left(c^\dagger_{k,\tau}\, c_{j,\sigma}\, \delta_{k,i}\, \delta_{\tau',\sigma} - c^\dagger_{i,\sigma}\, c_{k,\tau'}\, \delta_{k,j}\, \delta_{\tau,\sigma}\right. \tag{2.38}$$

$$\left. + c^\dagger_{k,\tau}\, c_{i,\sigma}\, \delta_{k,j}\, \delta_{\tau',\sigma} - c^\dagger_{j,\sigma}\, c_{k,\tau'}\, \delta_{k,i}\, \delta_{\tau,\sigma}\right) \tag{2.39}$$

$$= 2 \sum_i t_{i,k} \left(c^\dagger_{k,\tau}\, c_{i,\tau'} - c^\dagger_{i,\tau}\, c_{k,\tau'}\right)\,, \quad t_{i,k} = t_{k,i} \tag{2.40}$$

$$\sum_k \left[c^\dagger_{k,\tau}\, c_{k,\tau'}, H_t\right] = 0 \quad \Rightarrow [S_+, H_t] = [S_-, H_t] = [S_z, H_t] = 0\,. \tag{2.41}$$

As was expected from the construction, the kinetic term is SU(2)-invariant, provided that the hopping matrix $t_{i,j}$ is symmetric. We next treat the interaction term H_U:

$$\left[c^\dagger_{k,\tau}\, c_{k,\tau'}, c^\dagger_{i,\uparrow}\, c_{i,\uparrow}\, c^\dagger_{i,\downarrow}\, c_{i,\downarrow}\right] = \left[c^\dagger_{k,\tau}\, c_{k,\tau'}, c^\dagger_{i,\uparrow}\, c_{i,\uparrow}\right] c^\dagger_{i,\downarrow}\, c_{i,\downarrow} + c^\dagger_{i,\uparrow}\, c_{i,\uparrow} \left[c^\dagger_{k,\tau}\, c_{k,\tau'}, c^\dagger_{i,\downarrow}\, c_{i,\downarrow}\right] \tag{2.42}$$

$$= \left(c^\dagger_{k,\tau}\, c_{i,\uparrow}\, \delta_{k,i}\, \delta_{\tau',\uparrow} - c^\dagger_{i,\uparrow}\, c_{k,\tau'}\, \delta_{k,i}\, \delta_{\tau,\uparrow}\right) c^\dagger_{i,\downarrow}\, c_{i,\downarrow} \tag{2.43}$$

$$+ c^\dagger_{i,\uparrow}\, c_{i,\uparrow} \left(c^\dagger_{k,\tau}\, c_{i,\downarrow}\, \delta_{k,i}\, \delta_{\tau',\downarrow} - c^\dagger_{i,\downarrow}\, c_{k,\tau'}\, \delta_{k,i}\, \delta_{\tau,\downarrow}\right)\,. \tag{2.44}$$

By considering all cases of $\tau, \tau' = \uparrow, \downarrow$, one can easily verify that

$$\Rightarrow \sum_i \left[c^\dagger_{k,\tau}\, c_{k,\tau'}, c^\dagger_{i,\uparrow}\, c_{i,\uparrow}\, c^\dagger_{i,\downarrow}\, c_{i,\downarrow}\right] = 0\,, \tag{2.45}$$

which yields

$$[S_+, H_U] = [S_-, H_U] = [S_z, H_U] = 0\,. \tag{2.46}$$

For the density term H_μ, we calculate

$$\left[c^\dagger_{i,\tau}\, c_{i,\tau'}, c^\dagger_{j,\sigma}\, c_{j,\sigma}\right] = c^\dagger_{i,\tau}\, c_{j,\sigma}\, \delta_{i,j}\, \delta_{\tau',\sigma} - c^\dagger_{j,\sigma}\, c_{i,\tau'}\, \delta_{i,j}\, \delta_{\tau,\sigma} \tag{2.47}$$

$$\Rightarrow \sum_{\sigma,j} \left[c^\dagger_{i,\tau}\, c_{i,\tau'}, c^\dagger_{j,\sigma}\, c_{j,\sigma}\right] = 0\,, \tag{2.48}$$

which also leads to SU(2)-invariance:

$$[S_+, H_\mu] = [S_-, H_\mu] = [S_z, H_\mu] = 0\,. \tag{2.49}$$

Finally, we compute the following commutator

$$\sum_j \left[c^\dagger_{i,\tau}\, c_{i,\tau'}, c^\dagger_{j,\sigma}\, c_{j,\sigma}\right] = c^\dagger_{i,\tau}\, c_{i,\sigma}\, \delta_{\tau',\sigma} - c^\dagger_{i,\sigma}\, c_{i,\tau'}\, \delta_{\tau,\sigma} \tag{2.50}$$

$$\sum_j \left[c^\dagger_{i,\tau}\, c_{i,\tau'}, c^\dagger_{j,\uparrow}\, c_{j,\uparrow} - c^\dagger_{j,\downarrow}\, c_{j,\downarrow}\right] = c^\dagger_{i,\tau}\, c_{i,\uparrow}\, \delta_{\tau',\uparrow} - c^\dagger_{i,\uparrow}\, c_{i,\tau'}\, \delta_{\tau,\uparrow} \tag{2.51}$$

$$- c^\dagger_{i,\tau}\, c_{i,\downarrow}\, \delta_{\tau',\downarrow} + c^\dagger_{i,\downarrow}\, c_{i,\tau'}\, \delta_{\tau,\downarrow}\,, \tag{2.52}$$

which yields the commutators relevant to the magnetic-field-term H_B:

$$[S_z, H_B] = 0\,, \quad [S_+, H_B] = 2\,B\,S_+\,, \quad [S_-, H_B] = -2\,B\,S_-\,. \tag{2.53}$$

Unlike all other components of the Hamiltonian, H_B is not SU(2)-invariant. Thus, the Hubbard model exhibits a full SU(2)-spin symmetry if no magnetic field is applied to the system. For $|B| > 0$, SU(2) is broken and only S_z is conserved, yielding a U(1) symmetry.

Another possible choice of spinors that leads to the kinetic term H_t is

$$H_t = \sum_{\sigma,i,j} t_{i,j} \Psi^\dagger_{i,\sigma} \Psi_{j,\sigma}\,, \quad \Psi^\dagger_{i,\sigma} = \left(c^\dagger_{i,\sigma}, s_i\, c_{i,-\sigma}\right)^T\,, \quad \Psi_{i,\sigma} = \left(c_{i,\sigma}, s_i\, c^\dagger_{i,-\sigma}\right)^T\,, \tag{2.54}$$

where the sign s_i is defined to ensure normal ordering for all hopping partners, i.e. $s_i\, s_j = -1$. In a two-dimensional system, this only works out for a rectangular lattice with an even number of rows and columns and only nearest-heighbour hopping, yielding a bipartite lattice. This particular symmetry has thus a narrower applicability than the previous one, which only required a symmetric hopping matrix. The corresponding generators, denoted as C^i_+, C^i_-, and C^i_z, are calculated analogously to Eqs. (2.30) - (2.32):

$$C^i_+ = s_i\, c^\dagger_{i,\uparrow}\, c^\dagger_{i,\downarrow}\,, \quad C^i_- = s_i\, c_{i,\uparrow}\, c_{i,\downarrow}\,, \quad C^i_z = \frac{1}{2}\left(c^\dagger_{i,\uparrow}\, c_{i,\uparrow} + c^\dagger_{i,\downarrow}\, c_{i,\downarrow} - 1\right)\,, \tag{2.55}$$

$$C_+ = \sum_i C^i_+\,, \quad C_- = \sum_i C^i_-\,, \quad C_z = \sum_i C^i_z\,, \tag{2.56}$$

$$C_x = \frac{1}{2}\left(C_+ + C_-\right) , \quad C_y = \frac{1}{2i}\left(C_+ - C_-\right) . \tag{2.57}$$

The vector $\vec{C} = (C_x, C_y, C_z^i)^T$ is sometimes refered to as a pseudo-spin, leading to a charge-invariance- or η-pairing-symmetry. Note that C_+^i and C_-^i are basically the generator and annihilator of Cooper pairs, while C_z^i measures the deviation from half filling. To distinguish the previous symmetry from this one, we will refer to the former as $\mathrm{SU}(2)_{\mathrm{spin}}$ - symmetry, and the latter as $\mathrm{SU}(2)_{\mathrm{charge}}$ - symmetry.

Keeping in mind the restrictions on $t_{i,j}$, we now explicitly calculate the commutators of all components of the Hamiltonian to determine other constraints on $\mathrm{SU}(2)_{\mathrm{charge}}$. Since C_z is structurally similar to the generators of $\mathrm{SU}(2)_{\mathrm{spin}}$, one can immediately identify that

$$[C_z, H] = 0 , \tag{2.58}$$

which corresponds to the conservation of charge. For the commutators of C_+ and C_-, we first make note of another elementary commutation relation,

$$\left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, c_{i,\sigma}^\dagger\, c_{j,\sigma}\right] = c_{i,\sigma}^\dagger\, c_{k,\uparrow}^\dagger\, \delta_{j,k}\, \delta_{\downarrow,\sigma} - c_{i,\sigma}^\dagger\, c_{k,\downarrow}\, \delta_{j,k}\, \delta_{\uparrow,\sigma} , \tag{2.59}$$

and use it to calculate

$$\left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, H_t\right] = \sum_{\sigma,i,j} t_{i,j} \left(\left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, c_{i,\sigma}^\dagger\, c_{j,\sigma}\right] + \left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, c_{j,\sigma}^\dagger\, c_{i,\sigma}\right]\right) \tag{2.60}$$

$$= \sum_{\sigma,i,j} t_{i,j} \left(c_{i,\sigma}^\dagger\, c_{k,\uparrow}^\dagger\, \delta_{j,k}\, \delta_{\downarrow,\sigma} - c_{i,\sigma}^\dagger\, c_{k,\downarrow}\, \delta_{j,k}\, \delta_{\uparrow,\sigma}\right. \tag{2.61}$$

$$\left. + c_{j,\sigma}^\dagger\, c_{k,\uparrow}^\dagger\, \delta_{i,k}\, \delta_{\downarrow,\sigma} - c_{j,\sigma}^\dagger\, c_{k,\downarrow}\, \delta_{i,k}\, \delta_{\uparrow,\sigma}\right) \tag{2.62}$$

$$= 2 \sum_i t_{i,k} \left(c_{i,\downarrow}^\dagger c_{k,\uparrow}^\dagger - c_{i,\uparrow}^\dagger c_{k,\downarrow}^\dagger\right) , \tag{2.63}$$

$$\Rightarrow \sum_k \left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, H_t\right] = 0 , \quad \Rightarrow [C_+, H_t] = [C_-, H_t] = 0 . \tag{2.64}$$

This result is again expected, as it follows from the construction of the generators via spinors. The next step is to calculate the commutator with the summand of the interaction term H_U:

$$\left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, c_{i,\uparrow}^\dagger\, c_{i,\uparrow}\, c_{i,\downarrow}^\dagger\, c_{i,\downarrow}\right] = \left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, c_{i,\uparrow}^\dagger\, c_{i,\uparrow}\right] c_{i,\downarrow}^\dagger\, c_{i,\downarrow} + c_{i,\uparrow}^\dagger\, c_{i,\uparrow} \left[c_{k,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger, c_{i,\downarrow}^\dagger\, c_{i,\downarrow}\right] \tag{2.65}$$

$$= \left(c_{k,\uparrow}^\dagger\, c_{i,\uparrow}^\dagger\, \delta_{i,k}\, \delta_{\uparrow,\downarrow} - c_{i,\uparrow}^\dagger\, c_{k,\downarrow}^\dagger\, \delta_{i,k}\, \delta_{\uparrow,\uparrow}\right) c_{i,\downarrow}^\dagger\, c_{i,\downarrow} \tag{2.66}$$

$$+ c^\dagger_{i,\uparrow} c_{i,\uparrow} \left(c^\dagger_{i,\downarrow} c^\dagger_{k,\uparrow} \delta_{i,k} \delta_{\downarrow,\downarrow} - c^\dagger_{i,\downarrow} c^\dagger_{k,\downarrow} \delta_{i,k} \delta_{\uparrow,\downarrow} \right) \tag{2.67}$$

$$= -c^\dagger_{i,\uparrow} c^\dagger_{k,\downarrow} c^\dagger_{i,\downarrow} c_{i,\downarrow} \delta_{i,k} + c^\dagger_{i,\uparrow} c_{i,\uparrow} c^\dagger_{i,\downarrow} c^\dagger_{k,\uparrow} \delta_{i,k} \,. \tag{2.68}$$

The summation over i takes all lattice sites into account:

$$\sum_i s_k \left[c^\dagger_{k,\uparrow} c^\dagger_{k,\downarrow}, c^\dagger_{i,\uparrow} c_{i,\uparrow} c^\dagger_{i,\downarrow} c_{i,\downarrow} \right] = s_k \left(-c^\dagger_{k,\uparrow} c^\dagger_{k,\downarrow} c^\dagger_{k,\downarrow} c_{k,\uparrow} + c^\dagger_{k,\uparrow} c_{k,\downarrow} c^\dagger_{k,\downarrow} c^\dagger_{k,\uparrow} \right) \tag{2.69}$$

$$= -s_k \, c^\dagger_{k,\uparrow} c_{k,\downarrow} c^\dagger_{k,\uparrow} c^\dagger_{k,\downarrow} \tag{2.70}$$

$$= -s_k \, c^\dagger_{k,\uparrow} c^\dagger_{k,\downarrow} \tag{2.71}$$

$$= -C^k_+ \,. \tag{2.72}$$

$$\Rightarrow [C_+, H_U] = -U \, C_+ \quad [C_-, H_U] = U \, C_- \,. \tag{2.73}$$

Unlike in the case of $\mathrm{SU}(2)_{\mathrm{spin}}$, the ladder operators of $\mathrm{SU}(2)_{\mathrm{charge}}$ do not commute with the interaction term. For the remaining commutators, those with H_μ and H_B, we skip the intermediate steps, as they are completely analogous to the calculations above, and just state the final results, which can easily be verified:

$$[C_+, H_\mu] = 2 \, \mu \, C_+ \,, \quad [C_-, H_\mu] = -2 \, \mu \, C_- \,, \tag{2.74}$$

$$[C_+, H_B] = [C_-, H_B] = 0 \,. \tag{2.75}$$

Hence, a magnetic field does not disturb $\mathrm{SU}(2)_{\mathrm{charge}}$, while a chemical potential does. However, if we compare Eq. (2.73) with Eq. (2.74), we see that both commutators cancel each other out at half filling, i.e., $\mu = \frac{U}{2}$. In that case, one can add a physically irrelevant energy shift of $\frac{UN}{4}$ to the Hamiltonian, with N being the number of sites, and rewrite the interaction term as

$$H_U + H_\mu = U \sum_i \left(c^\dagger_{i,\uparrow} c_{i,\uparrow} - \frac{1}{2} \right) \left(c^\dagger_{i,\downarrow} c_{i,\downarrow} - \frac{1}{2} \right) \,. \tag{2.76}$$

In summary, $\mathrm{SU}(2)_{\mathrm{charge}}$ is only present for nearest-neighbor hopping and at half filling. Unfortunately, both of those conditions are violated in realistic models of high-temperature superconductivity. As was discussed extensively in the Introduction (Chapter 1), the most interesting doping regime is near half filling, while d-wave superconductors require next-nearest-neighbor hopping, yielding a lattice that is not bipartite. After all, $\mathrm{SU}(2)_{\mathrm{charge}}$ is exactly the symmetry which is broken by the generation of Cooper pairs.

We thus have to settle for particle-number conservation in Eq. (2.58). Since we have no interest in an external magnetic field, we can make use of the full spin-rotation symmetry, and arrive at $\mathrm{SU}(2)_{\mathrm{spin}} \otimes \mathrm{U}(1)_{\mathrm{charge}}$ as the symmetry group most suited to our purposes.

A more thorough examination of the symmetries of the Hubbard model is given in the textbook by Essler et al. [71].

2.3 Wigner-Eckart theorem

For a proper implementation of the SU(2)-symmetry, we first reiterate some of the algebraic properties of the angular momentum operator in quantum mechanics, $\vec{j} = (j_x, j_y, j_z)$, which includes half-integer spins. The individual Cartesian components are defined by the Lie algebra,

$$[j_k, j_l] = \mathrm{i}\, \epsilon_{klm}\, j_m \,, \tag{2.77}$$

which follows from the correspondence principle. Group theory dictates that, in order to take full advantage of the symmetry, we first form the Casimir operator, which is the square of the total momentum,

$$j^2 = \sum_i j_i^2 \,, \tag{2.78}$$

and the Cartan subalgebra, which is the largest subset of mutually commuting generators. For SU(2) or SO(3), this amounts to just one of the operators. By convention, one chooses j_z and expresses states in its eigenbasis, whereas the remaining components are fixed through Eq. (2.77). A state of angular momentum is thus uniquely identified by the total component, j, the z-component, m, and the equations

$$j_\pm \,|j\, m\rangle = \sqrt{j\,(j+1) - m\,(m \pm 1)}\,|j\, m \pm 1\rangle \;, \tag{2.79}$$

$$j_z \,|j\, m\rangle = m\,|j\, m\rangle \;, \tag{2.80}$$

with the ladder operators, $j_\pm = j_x \pm \mathrm{i}\, j_y$, and $|m| \leq j$. For detailed proofs, we refer to standard textbooks on elementary quantum mechanics, such as Ref. [72]. The generators of the Lie algebra can be used to carry out arbitrary rotations around the vectorial angle $\vec{\theta} = (\theta_x, \theta_y, \theta_z)$:

$$R = \exp\left(-\mathrm{i}\,\vec{\theta}\,\vec{j}\right) \,. \tag{2.81}$$

Due to Eq. (2.79) and Eq. (2.80), R can only map onto the same total angular momentum j; hence, the states $|j\ -j\rangle$, ... , $|j\, j\rangle$ span a $(2\,j+1)$-dimensional, invariant subspace.

Consider now the addition of two angular momenta, described by the operator

$$\vec{j} = \vec{j}_1 \otimes \mathbb{1}_2 + \mathbb{1}_1 \otimes \vec{j}_2 \,. \tag{2.82}$$

For given j_1 and j_2, the possible values of j are determined by the triangle inequality, $|j_1 - j_2| \leq j \leq j_1 + j_2$. The transition from the product basis, $|j_1\, m_1\rangle \otimes |j_2\, m_2\rangle = |j_1\, j_2\, m_1\, m_2\rangle$, to the full basis, $|j\, m\rangle$, is performed by inserting a resolution of unity,

$$|j\, m\rangle = \sum_{m_1, m_2} |j_1\, j_2\, m_1\, m_2\rangle \, \langle j_1\, j_2\, m_1\, m_2 | j\, m\rangle \ , \tag{2.83}$$

where the prefactors $\langle j_1\, j_2\, m_1\, m_2 | j\, m\rangle$ are called Clebsch-Gordan coefficients (CGCs). Their phase is arbitrary and thus usually is fixed by taking $\langle j_1\, j_2\, j_1\, m_2 | j\, j\rangle \in \mathbb{R}, > 0$. Furthermore, they satisfy the following orthogonality relations

$$\sum_{m_1, m_2} \langle j_1\, j_2\, m_1\, m_2 | j\, m\rangle \, \langle j_1\, j_2\, m_1\, m_2 | j'\, m'\rangle = \delta_{j,j'}\, \delta_{m,m'} \tag{2.84}$$

$$\sum_{j, m} \langle j_1\, j_2\, m_1\, m_2 | j\, m\rangle \, \langle j_1\, j_2\, m_1'\, m_2' | j\, m\rangle = \delta_{m_1, m_1'}\, \delta_{m_2, m_2'} \,. \tag{2.85}$$

With these prerequisites, the CGCs can be determined by applying

$$j_\pm = j_\pm^1 \otimes \mathbb{1}_2 + \mathbb{1}_1 \otimes j_\pm^2 \tag{2.86}$$

to both sides of Eq. (2.83):

$$\begin{aligned} j_\pm \, |j\, m\rangle = & \sum_{m_1', m_2'} j_\pm^1 \, |j_1\, j_2\, m_1'\, m_2'\rangle \, \langle j_1\, j_2\, m_1'\, m_2' | j\, m\rangle \\ & + \sum_{m_1', m_2'} j_\pm^2 \, |j_1\, j_2\, m_1'\, m_2'\rangle \, \langle j_1\, j_2\, m_1'\, m_2' | j\, m\rangle \ , \end{aligned} \tag{2.87}$$

$$\begin{aligned} & \sqrt{j\,(j+1) - m\,(m \pm 1)} \, |j\, m \pm 1\rangle \\ & = \sum_{m_1', m_2'} \sqrt{j_1\,(j_1+1) - m_1'\,(m_1' \pm 1)} \, |j_1\, j_2\, m_1' \pm 1\, m_2'\rangle \, \langle j_1\, j_2\, m_1'\, m_2' | j\, m\rangle \\ & \quad + \sum_{m_1', m_2'} \sqrt{j_2\,(j_2+1) - m_2'\,(m_2' \pm 1)} \, |j_1\, j_2\, m_1'\, m_2' \pm 1\rangle \, \langle j_1\, j_2\, m_1'\, m_2' | j\, m\rangle \ . \end{aligned} \tag{2.88}$$

Multiplying by $\langle j_1\, j_2\, m_1\, m_2|$ and using Eq. (2.84) yields the defining recursion relations:

$$\sqrt{j\,(j+1) - m\,(m \pm 1)} \, \langle j_1\, j_2\, m_1\, m_2 | j\, m \pm 1\rangle$$

$$= \sqrt{j_1\,(j_1+1) - m_1\,(m_1 \mp 1)}\,\langle j_1\, j_2\, m_1 \mp 1\, m_2 | j\, m\rangle$$
$$+ \sqrt{j_2\,(j_2+1) - m_2\,(m_2 \mp 1)}\,\langle j_1\, j_2\, m_1\, m_2 \mp 1 | j\, m\rangle\ . \tag{2.89}$$

For a given set of j_1, j_2, and j, we first set $m = j$ and choose the j_+-branch of Eq. (2.89):

$$\sqrt{j_1\,(j_1+1) - m_1\,(m_1 - 1)}\,\langle j_1\, j_2\, m_1 - 1\, m_2 | j\, m\rangle$$
$$= -\sqrt{j_2\,(j_2+1) - m_2\,(m_2 - 1)}\,\langle j_1\, j_2\, m_1\, m_2 - 1 | j\, m\rangle\ . \tag{2.90}$$

One can then start with $m_1 - 1 = j_1$ and $m_2 = m - m_1$ and express all coefficients as multiples of $\langle j_1\, j_2\, j_1\ (j - j_1) | j\, j\rangle$. The pending prefactor is fixed by the orthogonality relation in Eq. (2.84):

$$\sum_{m_1, m_2} \langle j_1\, j_2\, m_1\, m_2 | j\, j\rangle^2 = 1\,. \tag{2.91}$$

The results from $m = j$ can then be used to determine the CGCs for $m = j - 1$, which themselves are needed for $m = j - 2$, and so forth.

We are now prepared to examine operators. A scalar operator O, such as a Hamiltonian, is SU(2)-invariant if $R\,O\,R^{-1} = O$, where R is the rotation from Eq. (2.81). This leads to the following conservation law

$$\langle j'\, m' |\, O\, | j\, m\rangle \propto \delta_{j',j}\,\delta_{m',m}\,, \tag{2.92}$$

which is a special case of a block-sparse matrix, as already discussed in Sec. 2.2. A vector operator T fulfills SU(2)-symmetry if the following condition holds:

$$R\,T^k_q\,R^{-1} = \sum_{q'} T^k_{q'}\,R^k_{q',q}\,, \quad R^k_{q',q} = \langle k\, q' |\, R\, | k\, q\rangle\ . \tag{2.93}$$

The indices k and q are to be identified with a total angular momentum and with its z-component, respectively. As a whole, T^k_q can be seen as a rank-3 tensor, which, if rotated on all three legs around the same angle, is invariant. To determine the resulting conservation laws, we first derive some elementary commutation relations by considering infinitesimal rotations. First, we set $\theta_x = \theta_y = 0$, and $\theta_z \ll 1$ in Eq. (2.81) and get

$$(1 - \mathrm{i}\,\theta_z\, j_z)\ T^k_q\ (1 + \mathrm{i}\,\theta_z\, j_z) = \sum_{q'} T^k_{q'}\,\langle k\, q' |\,(1 - \mathrm{i}\,\theta_z\, j_z)\, | k\, q\rangle\ , \tag{2.94}$$

$$T^k_q + \mathrm{i}\,\theta_z\, T^k_q\, j_z - \mathrm{i}\,\theta_z\, j_z\, T^k_q + \mathcal{O}\left(\theta_z^2\right) = T^k_q - \mathrm{i}\,\theta_z\, q\, T^k_q\,, \tag{2.95}$$

$$\left[j_z, T^k_q\right] = q\, T^k_q\,. \tag{2.96}$$

For $\theta_y = \pm \mathrm{i}\theta_x$, $\theta_z = 0$ and $\theta_x \ll 1$, one obtains

$$(1 - \mathrm{i}\,\theta_x\, j_\pm)\; T_q^k\; (1 + \mathrm{i}\,\theta_x\, j_\pm) = \sum_{q'} T_{q'}^k\, \langle k\, q'|\, (1 - \mathrm{i}\,\theta_x\, j_\pm)\, |k\, q\rangle\ , \tag{2.97}$$

$$T_q^k + \mathrm{i}\,\theta_x\, T_q^k\, j_\pm - \mathrm{i}\,\theta_x\, j_\pm\, T_q^k + \mathcal{O}\left(\theta_x^2\right) = T_q^k - \mathrm{i}\,\theta_x \sqrt{k\,(k+1) - q\,(q \pm 1)}\, T_{q\pm1}^k\,, \tag{2.98}$$

$$\left[j_\pm, T_q^k\right] = \sqrt{k\,(k+1) - q\,(q \pm 1)}\, T_{q\pm1}^k\,. \tag{2.99}$$

The next step is to add an additional index α (describing a degnerate subspace) to each angular momentum and examine the properties of $T_q^k\, |\alpha\, j\, m\rangle$:

$$\begin{aligned} j_\pm\, T_q^k\, |\alpha'\, j'\, m'\rangle &= \left[j_\pm, T_q^k\right] |\alpha'\, j'\, m'\rangle + T_q^k\, j_\pm\, |\alpha'\, j'\, m'\rangle \\ &= \sqrt{k\,(k+1) - q\,(q \pm 1)}\, T_{q\pm1}^k\, |\alpha'\, j'\, m'\rangle \\ &\quad + \sqrt{j'\,(j'+1) - m'\,(m' \pm 1)}\, T_q^k\, |\alpha'\, j'\, m' \pm 1\rangle\ , \end{aligned} \tag{2.100}$$

$$\begin{aligned} j_z\, T_q^k\, |\alpha'\, j'\, m'\rangle &= \left[j_z, T_q^k\right] |\alpha'\, j'\, m'\rangle + T_q^k\, j_z\, |\alpha'\, j'\, m'\rangle \\ &= q\, T_q^k\, |\alpha'\, j'\, m'\rangle + T_q^k\, m'\, |\alpha'\, j'\, m'\rangle\ . \end{aligned} \tag{2.101}$$

Eq. (2.100) is structurally similar to the recursion relation in Eq. (2.89), which is why we define the following auxilliary vector

$$|\beta\, j''\, m''\rangle = \sum_{m',q} T_q^k\ |\alpha'\, j'\, m'\rangle\, \langle j'\, k\, m'\, q | j''\, m''\rangle\ , \tag{2.102}$$

and again apply both $j_\pm$ and j_z, which, after a few steps, yields

$$j_\pm\, |\beta\, j''\, m''\rangle = \sqrt{j''\,(j''+1) - m''\,(m'' \pm 1)}\, |\beta\, j''\, m''\rangle \tag{2.103}$$

$$j_z\, |\beta\, j''\, m''\rangle = m''\, |\beta\, j''\, m''\rangle\ . \tag{2.104}$$

The state $|\beta\, j''\, m''\rangle$ is thus an eigenfunction of the angular momentum operator. Returning to Eq. (2.102), we can recover $T_q^k\, |\alpha\, j\, m\rangle$ through the orthogonality relations of the CGCs:

$$\begin{aligned} &\sum_{j'',m''} |\beta\, j''\, m''\rangle\, \langle j'\, k\, m\, q' | j''\, m''\rangle \\ &= \sum_{m',q} \sum_{j'',m''} T_q^k\, |\alpha'\, j'\, m'\rangle\, \langle j'\, k\, m'\, q | j''\, m''\rangle\, \langle j'\, k\, m\, q' | j''\, m''\rangle \\ &= \sum_{m',q} T_q^k\, |\alpha'\, j'\, m'\rangle\ \delta_{m,m'}\, \delta_{q,q'} \end{aligned} \tag{2.105}$$

$$= T^k_{q'} \left|\alpha' \, j' \, m\right\rangle \ , \tag{2.106}$$

$$\Rightarrow \ T^k_q \left|\alpha' \, j' \, m'\right\rangle = \sum_{j'',m''} \left|\beta \, j'' \, m''\right\rangle \left\langle j' \, k \, m' \, q | j'' \, m''\right\rangle \ . \tag{2.107}$$

Applying $\langle \alpha \, j \, m|$ from the left gives the matrix elements of T^k_q

$$\begin{aligned} \left\langle \alpha \, j \, m\right| T^k_q \left|\alpha' \, j' \, m'\right\rangle &= \sum_{j'',m''} \left\langle \alpha \, j \, m | \beta \, j'' \, m''\right\rangle \left\langle j' \, k \, m' \, q | j'' \, m''\right\rangle \\ &= \left\langle \alpha \, j \, m | \beta \, j \, m\right\rangle \left\langle j' \, k \, m' \, q | j \, m\right\rangle \ . \end{aligned} \tag{2.108}$$

We are thus left with a CGC, $\langle j' \, k \, m' \, q | j \, m\rangle$, and the scalar product $\langle \alpha \, j \, m | \beta \, j \, m\rangle$ with an unkown parameter β. Since it is independent of any z-component of an angular momentum due to

$$\begin{aligned} \left\langle \alpha \, j \, m | \beta \, j \, m\right\rangle &= \left(\sqrt{j \, (j+1) - m \, (m-1)}\right)^{-1} \left\langle \alpha \, j \, m\right| \left(j_+ \left|\beta \, j \, m - 1\right\rangle\right) \\ &= \left(\sqrt{j \, (j+1) - m \, (m-1)}\right)^{-1} \left(j_- \left|\alpha \, j \, m\right\rangle\right)^\dagger \left|\beta \, j \, m - 1\right\rangle \\ &= \left\langle \alpha \, j \, m - 1 | \beta \, j \, m - 1\right\rangle \ , \end{aligned} \tag{2.109}$$

the most general object that relates it to the left-hand side of Eq. (2.108) is the reduced matrix element $\left\langle \alpha \, j \| T^k \| \alpha' \, j'\right\rangle$, which finally leads us to the *Wigner-Eckart theorem*

$$\left\langle \alpha \, j \, m\right| T^k_q \left|\alpha' \, j' \, m'\right\rangle = \left\langle \alpha \, j \| T^k \| \alpha' \, j'\right\rangle \left\langle j' \, k \, m' \, q | j \, m\right\rangle \ . \tag{2.110}$$

Its advantage is twofold: First, the selection rules $q = m - m'$ and $|j - j'| \leq k \leq j + j'$, which are dictated by the CGC on the right-hand side, narrow down the matrix elements one has to take into account. Second, given a non-zero element of T^k_q, all other elements that differ only in m, m', and q are fixed by the SU(2) symmetry. One of the most popular applications of the Wigner-Eckart theorem is the calculation of transition amplitudes between orbitals in the hydrogen atom. For this purpose, Eq. (2.110) determines which transitions are allowed purely based on symmetry considerations and relates them to each other through known CGCs, which can be calculated in advance. In this thesis, the Wigner-Eckart theorem serves as a stepping stone for SU(2)-invariant tensors with arbitrary rank.

The line of reasoning in this chapter is in large part based on the textbook by Messiah [72].

3 Quantum Entanglement

3.1 AKLT-model

One of the first successful attempts at using entanglement to describe ground states was made by Affleck et al. [40] in the form of valence-bond solid states, also known as AKLT states, named after their inventors. While the original paper emphasized the relationship between finite gaps and exponentially decaying correlation functions in integer spin systems, as had been predicted by Haldane [73–75], we are going to focus on the unconventional way of storing wave functions, which will serve as an introduction to tensor network states and hence the main subject of this thesis.

Consider a spin-1 chain, where each site can assume one of the three configurations $S_z = 1$, $S_z = 0$, and $S_z = -1$. The central idea is to forge a quantum state by reinterpreting each site as a triplet of two spin-1/2 particles,

$$|S = 1, S_z = 1\rangle = |\uparrow\uparrow\rangle \ , \tag{3.1}$$

$$|S = 1, S_z = 0\rangle = \frac{1}{\sqrt{2}} \left(|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle\right) \ , \tag{3.2}$$

$$|S = 1, S_z = -1\rangle = |\downarrow\downarrow\rangle \ , \tag{3.3}$$

and connect them through a singlet

$$|S = 0, S_z = 0\rangle = \frac{1}{\sqrt{2}} \left(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle\right) \tag{3.4}$$

across the triplet boundary, as shown in Fig. 3.1.

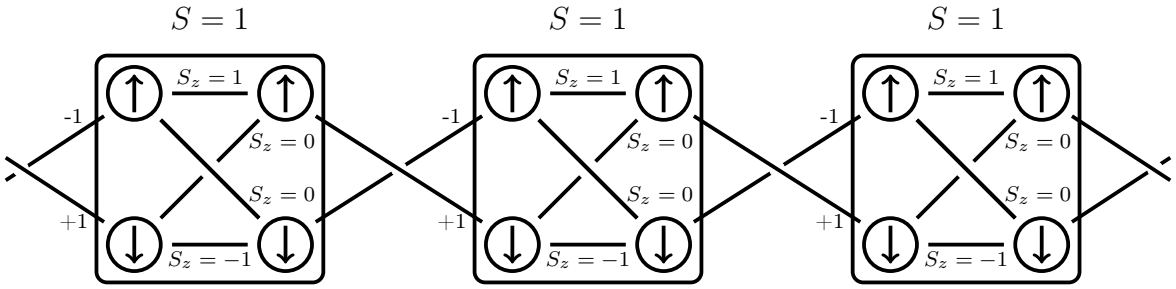


Figure 3.1: Diagrammatic representation of an AKLT-state. The boxes enclose spin-1 sites, whereas the lines crossing their boundaries connect them through singlets.

Chemically, this can be understood as two spin-one particles being connected by a valence bond. In this way, the overall state is guaranteed to have spin-zero and hence be in a low-energy regime. Each site participates in four different product states $|\uparrow\uparrow\rangle$, $\frac{1}{\sqrt{2}}|\uparrow\downarrow\rangle$, $\frac{1}{\sqrt{2}}|\downarrow\uparrow\rangle$ and $|\downarrow\downarrow\rangle$, which means that $|S = 1, S_z = 0\rangle$ is fragmented into its two parts. The

AKLT-state on N sites reads

$$|\text{AKLT}\rangle_N = \psi_{a_1,b_1} |a_1\, b_1\rangle_1 \cdot \varepsilon_{b_1,a_2} \cdot \psi_{a_2,b_2} |a_2\, b_2\rangle_2 \cdot \varepsilon_{b_2,a_3} \cdot \ldots \cdot \varepsilon_{b_{N-1},a_N} \cdot \psi_{a_N,b_N} |a_N\, b_N\rangle_N \ , \tag{3.5}$$

with $a_i = \uparrow, \downarrow$, $b_i = \uparrow, \downarrow$, $\psi_{\uparrow,\uparrow} = \psi_{\downarrow,\downarrow} = 1$, $\psi_{\uparrow,\downarrow} = \psi_{\downarrow,\uparrow} = \frac{1}{\sqrt{2}}$, $\varepsilon_{\uparrow,\downarrow} = -\varepsilon_{\downarrow,\uparrow} = 1$, and $\varepsilon_{\uparrow,\uparrow} = \varepsilon_{\downarrow,\downarrow} = 0$. To retrieve the original spin-1 basis, we define the local projector

$$P = |1\rangle \langle\uparrow\uparrow| + \frac{1}{\sqrt{2}} \left(|0\rangle \langle\uparrow\downarrow| + |0\rangle \langle\downarrow\uparrow|\right) + |-1\rangle \langle\downarrow\downarrow| \ , \tag{3.6}$$

with $|1\rangle \equiv |S = 1, S_z = 1\rangle$, $|0\rangle \equiv |S = 1, S_z = 0\rangle$, and $|-1\rangle \equiv |S = 1, S_z = -1\rangle$. Consider now the following operator introduced by Affleck et al. [40],

$$H_i = \frac{1}{2} \vec{S}_i \, \vec{S}_{i+1} + \frac{1}{6} \left(\vec{S}_i \, \vec{S}_{i+1}\right)^2 + \frac{1}{3} \, , \tag{3.7}$$

with $\vec{S}_i = \left(S_x^i, S_y^i, S_z^i\right)^T$ and

$$S_x = \frac{1}{\sqrt{2}} \left(|1\rangle \langle 0| + |0\rangle \langle 1| + |0\rangle \langle -1| + |-1\rangle \langle 0|\right) \ , \tag{3.8}$$

$$S_y = \frac{i}{\sqrt{2}} \left(-|1\rangle \langle 0| + |0\rangle \langle 1| - |0\rangle \langle -1| + |-1\rangle \langle 0|\right) \ , \tag{3.9}$$

$$S_z = |1\rangle \langle 1| - |-1\rangle \langle -1| \ . \tag{3.10}$$

It can be shown that H_i is a projector onto the spin-2 subspace of two spin-1 particles, hence, $H_i^2 = H_i$. If we now take an AKLT-state for $N = 2$,

$$|\text{AKLT}\rangle_2 = \psi_{a_i,b_i} |a_i\, b_i\rangle_i \cdot \varepsilon_{b_i,a_{i+1}} \cdot \psi_{a_{i+1},b_{i+1}} |a_{i+1}\, b_{i+1}\rangle_{i+1} \ , \tag{3.11}$$

and apply the local projector from Eq. (3.6) on both sites, followed by H_i, we find

$$H_i \, P_i \, P_{i+1} \, |\text{AKLT}\rangle_2 = 0 \, . \tag{3.12}$$

Since projectors cannot have negative eigenvalues, $|\text{AKLT}\rangle_2$ is the ground state of H_i, which makes $|\text{AKLT}\rangle_N$ the ground state of both H_i and H_{AKLT} with

$$H_{AKLT} = \sum_{i=1}^{N-1} H_i = \sum_{i=1}^{N-1} \left(\frac{1}{2} \vec{S}_i \, \vec{S}_{i+1} + \frac{1}{6} \left(\vec{S}_i \, \vec{S}_{i+1}\right)^2 + \frac{1}{3}\right) \ . \tag{3.13}$$

For our purposes, the important result of this exercise is that we have found a Hamiltonian whose ground state can be stored exactly and scalably. Usually, the size of a many-body quantum state grows exponentially with system size, whereas Eq. (3.5) grows linearly. The

state $|\text{AKLT}\rangle$ can thus be viewed as a proto-tensor network state, with $|a_i, b_i\rangle$ representing physical bonds, where the two singlets in between constitute a virtual bond, which encodes quantum mechanical entanglement [76].

3.2 Von Neumann entropy

The entanglement utilized in the previous chapter can be quantified by the von Neumann entropy, which we shall now motivate, following Preskill [77]. Consider an alphabet consisting of k letters y_1,...,y_k, and the respective probabilties $p(y_1)$,...,$p(y_k)$ with which they occur in a certain language. Due to the independence of the $p(x_i)$, the probabilty of an n-letter word is given by

$$p(x_1, ..., x_n) = \prod_{i=1}^{n} p(x_i) \ . \tag{3.14}$$

Clearly, the number of possibilities grows exponentially with word size.

Now we ask if it is possible to take advantage of the probabilities, which are inherent to a language, in order to compress messages. Because of the law of large numbers, the number of different messages for $n \gg 1$ is given by the generalized binomial coefficient

$$N = \frac{n!}{\prod_{i=1}^{k} (np(y_i))!} \, . \tag{3.15}$$

The Stirling approximation $\ln n! \approx n \ln n - n$ yields

$$\ln N \approx n \ln n - n - \sum_{i=1}^{k} (np(y_i) \ln (np(y_i)) - np(y_i)) \tag{3.16}$$

$$= n \ln n - \sum_{i=1}^{k} np(y_i) \ln (np(y_i)) \tag{3.17}$$

$$= -n \sum_{i=1}^{k} p(y_i) \ln (p(y_i)) \, , \tag{3.18}$$

where we have used the normalization of probabilities twice. The *Shannon entropy* [78] is now defined as

$$H(p) = -\sum_{i=1}^{k} p(y_i) \ln (p(y_i)), \tag{3.19}$$

which leads to the following expression for the number of attainable messages:

$$N = \mathrm{e}^{n\,H(p)}\,. \tag{3.20}$$

The range of $H\,(p)$ and thus of N is determined by the worst-case scenario, which is an equal distribution of probabilities,

$$H_{max}\,(p) = -\sum_{i=1}^{k} \frac{1}{k} \ln \frac{1}{k} = \ln k \quad \Rightarrow \quad N_{max} = k^n\,, \tag{3.21}$$

and by the best case scenario, which is the occurence of just one letter:

$$H_{min}\,(p) = 0 \quad \Rightarrow \quad N_{min} = 1\,. \tag{3.22}$$

We conclude that, unless Eq. (3.21) holds, actually using n letters to store an n-letter message is wasteful, as the number of possible configurations is much smaller than k^n. Alternatively, one could say that, once a certain subset $m < n$ of letters is known, the rest of the word can be inferred through $H\,(p)$ and is, in fact, redundant. The Shannon entropy can hence be viewed as a measure for knowledge or for ignorance, depending on the point of view.

The von Neumann entropy applies this idea to quantum mechanics. Suppose we have a generic, pure quantum state on N sites, which we conceptualize as a full message:

$$|\psi\rangle = \sum_{j_1,...,j_N} \psi_{j_1,...,j_N}\,|j_1,...,j_N\rangle\ . \tag{3.23}$$

Here the $|j_1,...,j_N\rangle$ are orthonormal basis states in the total Hilbert space $\mathcal{H} = \mathcal{H}_1 \otimes ... \otimes \mathcal{H}_N$, and $\psi_{j_1,...,j_N}$ are coefficients. If the system is bipartite, we can split it into $A = [1,...,i]$ and $B = [i+1,...,N]$, which yields the following form of the wave function

$$|\psi\rangle = \sum_{a,b} \psi_{a,b}\,|\psi_a\rangle \otimes |\psi_b\rangle\ , \tag{3.24}$$

where $a = \{j_1,...,j_i\}$, $b = \{j_{i+1},...,j_N\}$, $|\psi_a\rangle = |j_1,...,j_i\rangle$, and $|\psi_b\rangle = |j_{i+1},...,j_N\rangle$. A measurement on one of the subsystems, say A, is now performed, yielding the expectation value

$$\langle\psi|\,\hat{O}_A \otimes \mathbb{1}_B\,|\psi\rangle = \sum_{a,b,a',b'} \psi^*_{a,b}\psi_{a',b'}\,\langle\psi_a|\,\hat{O}_A\,|\psi_{a'}\rangle\,\langle\psi_b|\psi_{b'}\rangle \tag{3.25}$$

$$= \sum_{a,a',b} \psi^*_{a,b}\psi_{a',b}\,\langle\psi_a|\,\hat{O}_A\,|\psi_{a'}\rangle\ . \tag{3.26}$$

If we define the *reduced density matrix* of subsystem A as

$$\rho_A = \mathrm{Tr}_B \left(|\psi\rangle \langle\psi| \right) = \sum_b \langle\psi_b|\psi\rangle \langle\psi|\psi_b\rangle = \sum_{a,a',b} \psi^*_{a,b}\psi_{a',b} |\psi_a\rangle \langle\psi_{a'}| \,, \tag{3.27}$$

the measurement of $\hat{O}_A$ reads

$$\langle\psi| \hat{O}_A \otimes \mathbb{1}_B |\psi\rangle = \mathrm{Tr}_A \left(\rho_A \hat{O}_A \right) . \tag{3.28}$$

Compared to the full state $|\psi\rangle$, the density matrix ρ_A can be seen as a partial message, whose eigenstates $|i\rangle$ and eigenvalues ρ_i can be identified with letters and probabilities, respectively. Eq. (3.19) then yields the *von Neumann entropy*

$$S = -\sum_{i=1}^{D} \rho_i \ln \rho_i \,, \qquad 0 \leq \rho_i \leq 1 \,, \qquad \sum_{i=1}^{D} \rho_i = 1 \,, \tag{3.29}$$

which is a measure for the quantum mechanical entanglement between subsystem A and its complement B. For $S = 0$, ρ_A and, consequentially, ρ_B both consist of exactly one state, which means there is no entanglement at all, the full state $|\psi\rangle$ can be written as a product state, and both subsystems can be measured independently. For $S > 0$, A and B are entangled, and measurement in one of the two systems will affect the other.

The idea of tensor networks is to utilize the knowledge about B that is already stored within A by deducing the number of states D starting from the entropy. For a given D, the upper bound for the entropy is given by the full entanglement,

$$S \leq -\sum_{i=1}^{D} \frac{1}{D} \ln \frac{1}{D} = \ln D, \tag{3.30}$$

which, in turn, means that the number of states necessary to express a wave function faithfully grows exponentially with entropy. If, however, some cases exhibit a behaviour in which S grows slowly with system size or is even constant, it would be useful to express $|\psi\rangle$ through multiple density-matrix-like objects. Tensor networks can thus be viewed as a means of data compression for quantum mechanics.

3.3 Area law

We now relate the von Neumann entropy to actual physical models and examine its scaling with respect to system size. Suppose a wave function $|\psi\rangle$ is split into two subsystems A and B, as depicted in Fig. 3.2. Intuitively, one would assume a volume law, meaning that the entropy of A, $S(A)$, is proportional to its volume, as it is actually the case for thermal states. However, some ground states and other atypical states follow an *area law*, which

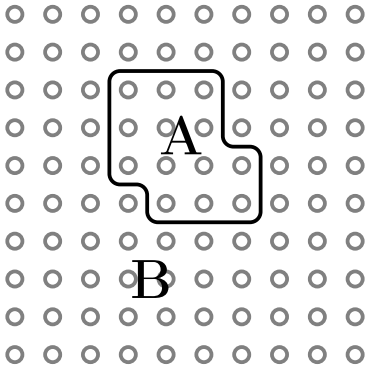


Figure 3.2: Wave function on a 10x10 lattice, split into subsystems A and B.

is to say that

$$S(A) \propto \partial A, \tag{3.31}$$

where ∂A is the size of the boundary of A. In this case, the number of mixed states is constant with respect to system size due to Eq. (3.30), and the size of the overall wave function grows linearly. The wave function $|\psi\rangle$ thus lives in a tiny subsector of the exponentially large Hilbert space. This particular relationship is one of the main inspirations for tensor network representations, as they obey an area law by construction, which we will explain in subsequent chapters.

Mathematical proofs of the entropy scaling in Hamiltonians are cumbersome and exceed the scope of this thesis. Here we will simply enumerate some models that have been examined for their entanglement entropy and are mentioned in the review on area laws by Eisert et al. [79]:

The first area law for lattice systems was proven by Audenaert et al. [80] for the ground state of the harmonic chain on L sites with periodic boundary conditions

$$H = \frac{1}{2}\sum_{i=1}^{L}\left(p_i^2 + a\,x_i^2 + 2\,b\,x_i\,x_{i+1}\right)\,, \tag{3.32}$$

where x_i and p_j are canonical operators, a is an onsite term, and b is the coupling between nearest neighbors. Key to the proof is the logarithmic negativity E [81], which is an upper bound for the entropy, and, after lengthy calculations, turns out to be

$$E = \frac{1}{4}\log_2\left(\frac{a+2\,|b|}{a-2\,|b|}\right) = \frac{1}{2}\log_2\left(\frac{\sqrt{||X||}}{\Delta E}\right). \tag{3.33}$$

Here X is the circulant matrix that arises if we write the x_i-dependence in Eq. (3.32) as $x_i\,X_{i,j}\,x_j$, $||X||$ is its operator norm, and $\Delta E = \sqrt{a-2\,|b|}$ is the *energy gap*, which is the difference between the ground state energy and the energy of the first excited state. Since

there is no dependence on L and the boundary of a mixed state on a chain is constant, Eq. (3.33) yields an area law as long as the Hamiltonian is gapped.

The Klein-Gordon-field Hamiltonian can be constructed by taking the continuous limit of the harmonic chain. If we substitute $a = m^2 + 2\,N^2$, $b = -N^2$, $i = x\,N$, $x_i = \sqrt{N}\phi\,(x)$ and $p_i = \sqrt{N}\pi\,(x)$ in Eq. (3.32), where m is the mass, $\phi\,(x)$ and $\pi\,(x)$ are canonical variables, x is a continuous position, and N is the number of sites, we obtain

$$H = \frac{1}{2}\int_0^1 dx\,\left[\pi^2\,(x) + \left(\frac{\partial}{\partial x}\phi\,(x)\right)^2 + m^2\phi^2\,(x)\right] \tag{3.34}$$

in the $N \to \infty$ limit. Using the same substitutions as in Eq. (3.33) gives the upper bound

$$E = \lim_{N\to\infty}\frac{1}{2}\log_2\left(\frac{2\,N}{m}\right). \tag{3.35}$$

for the entropy. Here the area law is violated, and the entropy diverges, which can be related to a vanishing energy gap.

Consider now quasi-free fermions on a chain of N sites described by the Hamiltonian

$$H = \lambda\sum_{i=1}^{N}\left(f_i^\dagger f_i - \frac{1}{2}\right) + \frac{1}{4}\sum_{\langle i,j\rangle}\left[f_i^\dagger f_j - f_i f_j^\dagger + \mathrm{sgn}\,(i-j)\,\gamma\left(f_i f_j - f_i^\dagger f_j^\dagger\right)\right], \tag{3.36}$$

where λ and γ are model parameters, $\langle i,j\rangle$ denotes nearest neighbors, and $f_i^\dagger$ and f_i are fermionic creation and annihilation operators, respectively. Application of the Jordan-Wigner transformation converts this model to the XY spin model:

$$H = -\frac{1}{2}\sum_{\langle i,j\rangle}\left(\frac{1+\gamma}{4}\sigma_i^x\sigma_j^x + \frac{1-\gamma}{4}\sigma_i^y\sigma_j^y\right) - \frac{\lambda}{2}\sum_{i=1}^{N}\sigma_i^z\,, \tag{3.37}$$

with the σ^x, σ^y and σ^z Pauli matrices. Because of translational invariance, the energy spectrum can be calculated using Fourier transformation and reads

$$E\,(k) = \sqrt{\left[\lambda - \cos\left(\frac{2\,\pi\,k}{N}\right)\right]^2 + \gamma^2\sin^2\left(\frac{2\,\pi\,k}{N}\right)}\,, \tag{3.38}$$

where $k = 1, ..., N$. Just as in the case of the harmonic chain, entropy scaling depends on whether the Hamiltonian is gapped or not. For $|\lambda| = 1$ or $\gamma = 0$ and $|\lambda| \leq 1$, the system is critical, which means the energy gap vanishes in the thermodynamic limit and the entropy scales as

$$S \propto \log_2 N\,. \tag{3.39}$$

Otherwise, the energy gap is greater than zero and the area law is fulfilled. For detailed derivations, see Refs. [82–84].

A major contribution was made by Hastings [85], who considered the ground states of gapped, local, one-dimensional Hamiltonians H in general. In this case, the entropy S has the upper bound

$$S \leq c_0 \, \xi' \ln(\xi') \ln(D) 2^{\xi' \ln(D)} , \quad \xi = \max\left(\frac{2\,v}{\Delta E}, \xi_C\right) , \quad \xi' = 6\xi \, . \tag{3.40}$$

Here D is the dimension of the local Hilbert space of each site, ΔE is the energy gap, and c_0 is a numerical constant. The velocity v and correlation length ξ_C are chosen so that the Lieb-Robinson bound [86]

$$||[A(t), B]|| \leq c \cdot |X| \, ||A|| \, ||B|| \, . \exp\{[-\xi_C \operatorname{dist}(X, Y)]\} \tag{3.41}$$

is fulfilled. The arbitrary operators A and B have support on the sets of Hilbert spaces X and Y, and $A(t) = \exp\{(i\,H\,t)\}\, A \, \exp\{(-i\,H\,t)\}$. The distance between the two sets is given by $\operatorname{dist}(X, Y) = \min_{i \in X, j \in Y}(|i - j|)$ and $|t| = l/v$, where l is the extension of X. Since we again find no dependence on the system size, Eq. (3.40) yields an area law.

The last one-dimensional model mentioned by Eisert et al. [79] is a disordered spin chain, described by the Hamiltonian

$$H = \sum_{i=1}^{L} J_i \left(\sigma_i^x \sigma_{i+1}^x + \sigma_i^y \sigma_{i+1}^y + \sigma_i^z \sigma_{i+1}^z\right) , \tag{3.42}$$

where J_i are chosen from some continuous distribution. Refael and Moore [87, 88] showed using the real-space renormalization group that the entropy scaling is similar to that of the harmonic chain on N sites:

$$S \propto \log_2(N) \, . \tag{3.43}$$

While the entropy scaling in one-dimensional systems is well understood, definite statements for ground states on two-dimensional lattices are scarce due to the more complicated structure of the boundary. One of the only existing analytic results is with respect to quasi-free bosons and fermions, which can be quickly summarized: Just like in the 1D case, the ground state exhibits an area law as long as the Hamiltonian is gapped [89, 90]. For critical fermionic systems, the area law is violated by logarithmic corrections with respect to system size [91]. In recent years, some progess has been made in determining the entanglement of interacting fermions in two dimensions numerically. In particular, Assaad and Toldin used Quantum Monte Carlo (QMC) simulations to calculate the Renyi entropy [92–95], which is a generalization of the von Neumann entropy. However, a general, analytic statement for the entropy of local Hamiltonians in two dimensions is an

open problem which is one of the major challenges in the field of quantum information.

4 Symmetric tensors

4.1 Definition

Tensors are the elementary building blocks of tensor networks and can be part of a wave function, part of a Hamiltonian, or can result from an intermediate contraction. We define them as as a map from a tensor product of Hilbert spaces to a complex number [96]

$$T : \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \ ... \ \otimes \mathcal{H}_r \to \mathbb{C} \,, \tag{4.1}$$

where r is the rank. With respect to a specific basis, this translates to a multidimensional array $T_{i_1,i_2,...,i_r}$ [97], which can be drawn as a vertex with attached lines, as shown in Fig. 4.1.

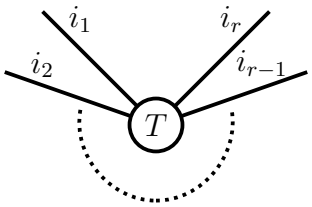
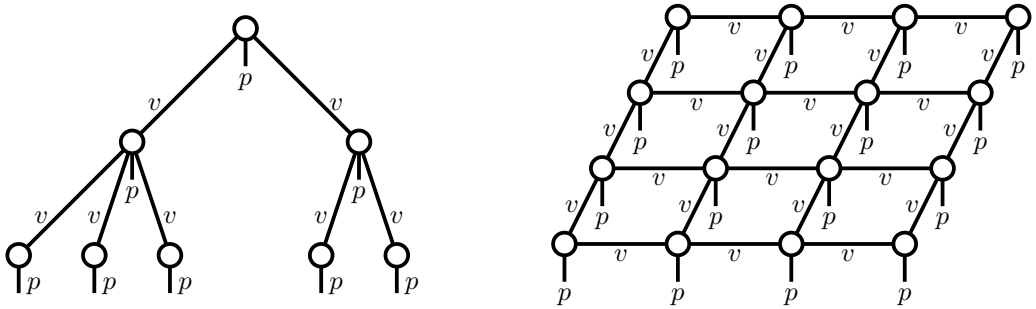


Figure 4.1: Rank-r tensor.

The $\mathcal{H}_i$ can be interpreted as bonds or indices and are either ingoing or outgoing, depending on whether the space is dual or not. Furthermore, they are either *physical*, in which case they act on a local site and are used to tap observables such as charge or spin, or *virtual*, in that they are used to connect adjacent physical sites, as is shown in Fig. 4.2. Virtual spaces arise as a consequence of an initalization or a decomposition of a previously contracted larger tensor. For acyclic tensor networks, i.e., tree-tensor networks (TTNs), the system is multipartite and the states in virtual spaces encode the entanglement between adjacent subsystems. In cyclic tensor networks, such as projected entangled pair states (PEPSs), virtual states cannot be so easily related to entanglement because there are many paths between two physical sites. Instead, the von-Neumann entropy of a subsystem must be calculated explicitly.

Tensor networks inherit the symmetries of the Hamiltonian under investigation by construction, which is why we implement the $\mathrm{SU}(2)_{\mathrm{spin}} \otimes \mathrm{U}(1)_{\mathrm{charge}}$ - group for the Hubbard model (see Sec. 2.2). Without symmetries, tensor network algorithms are virtually intractable due to the sheer size of the tensors, which grow exponentially with the rank. First, we consider a U(1)-symmetric rank-3 tensor, i.e., a vector operator T_q which satisfies the following condition

$$R\, T_q\, R^{-1} = \sum_{q'} T_{q'}\, R_{q',q} \,, \quad R = \exp\left(-i\,\theta\, Q\right) \,. \tag{4.2}$$



(a) Tree-tensor network

(b) Cyclic tensor network

Figure 4.2: Wave functions as tensor networks with physical bonds p and virtual bonds v.

One has to imagine R rotating all three bonds of T_q around the same angle θ. The Hermitian operator Q with eigenvalues q commutes with H and leads to the U(1)-symmetry. Due to Eq. (2.96), Eq. (4.2) is equivalent to

$$[Q, T_q] = q\, T_q\,. \tag{4.3}$$

For states $|\alpha\, m\rangle$ with $Q\,|\alpha\, m\rangle = m\,|\alpha\, m\rangle$, we find

$$Q\, T_q\ |\alpha\, m\rangle = [Q, T_q]\,|\alpha\, m\rangle + T_q\, Q\ |\alpha\, m\rangle \tag{4.4}$$

$$= (q + m)\ T_q\,|\alpha\, m\rangle\ , \tag{4.5}$$

$$\Rightarrow\ \langle \alpha'\, m'|\, T_q\,|\alpha\, m\rangle \propto \delta_{m',q+m}\,, \tag{4.6}$$

leading to the conservation law

$$m' = q + m\,. \tag{4.7}$$

By renaming α and α' into t_1 and t_2, adding a degeneracy t_3 to q, and renaming the quantum numbers as q_1, q_2 and q_3, one can also write

$$T_{(q_1 t_1),(q_2 t_2),(q_3 t_3)} = T_{(t_1 t_2 t_3),(q_1 q_2 q_3)}\, \delta_{q_1,q_2+q_3}\,, \tag{4.8}$$

which makes the symmetric tensor structure more apparent. Leaving out one or two bonds, or contracting multiple rank-3 tensors along one bond gives the generalization of the conservation law to tensors of arbitrary rank:

$$\sum_{k=1}^{r} s_k\, q_k = 0\,. \tag{4.9}$$

Here q_k is a quantum number on bond k, while s_k is defined as $+1$ for a Hilbert space and -1 for a corresponding dual space, or, equivalently, $+1$ for an ingoing bond and -1 for

an outgoing bond. Symmetric tensors T can thus be expressed as a direct sum of smaller blocks,

$$T = \bigoplus_{\{q_k\}} T_{\{q_k\}} \,, \tag{4.10}$$

where each set $\{q_k\}$ satisfies Eq. (4.9). The tensor T as a whole is the structural- or block-sparse tensor determined by the symmetries, while each $T_{\{q_k\}}$ is a degenerate- or dense tensor [98], on which level the actual processing will take place. For a rank-2 tensor, i.e., a matrix, Eq. (4.10) leads to Eq. (2.22) with one quantum number.

In order to determine the structure of SU(2)-invariant tensors, we proceed analogously to the previous case and start with the properties of a rank-3 tensor, which is covered by the Wigner-Eckart theorem (Eq. (2.110)), and, after renaming some indices, reads

$$T_{i_1 i_2 i_3} = T_{(j_1 m_1 t_1),(j_2 m_2 t_2),(j_3 m_3 t_3)} = P_{(j_1 t_1),(j_2 t_2),(j_3 t_3)}\, C_{(j_1 m_1),(j_2 m_2),(j_3 m_3)} \,. \tag{4.11}$$

The indices j_i, m_i, and t_i represent angular momentum, its z-component, and degeneracy, respectively, while C is a CGC. The tensor P is the reduced matrix element, amended by one dense index. Rank-1 and rank-2 tensors are again trivial special cases of rank-3, whereas higher ranks are more complicated. Suppose we have an SU(2)-invariant rank-4 tensor, which can be viewed as the contraction of two SU(2)-invariant rank-3 tensors with an internal index $i_a = (j_a, m_a, t_a)$:

$$\begin{aligned}
T_{i_1 i_2 i_3 i_4} &= T_{(j_1 m_1 t_1),(j_2 m_2 t_2),(j_3 m_3 t_3),(j_4 m_4 t_4)} \\
&= \sum_{j_a m_a t_a} T_{(j_1 m_1 t_1),(j_2 m_2 t_2),(j_a m_a t_a)}\, T_{(j_a m_a t_a),(j_3 m_3 t_3),(j_4 m_4 t_4)} \\
&= \sum_{j_a m_a t_a} P_{(j_1 t_1),(j_2 t_2),(j_a t_a)}\, C_{(j_1 m_1),(j_2 m_2),(j_a m_a)} \\
&\qquad\qquad \cdot P_{(j_a t_a),(j_3 t_3),(j_4 t_4)}\, C_{(j_a m_a),(j_3 m_3),(j_4 m_4)} \\
&= \sum_{j_a} \sum_{t_a} P_{(j_1 t_1),(j_2 t_2),(j_a t_a)}\, P_{(j_a t_a),(j_3 t_3),(j_4 t_4)} \\
&\qquad\qquad \cdot \sum_{m_a} C_{(j_1 m_1),(j_2 m_2),(j_a m_a)}\, C_{(j_a m_a),(j_3 m_3),(j_4 m_4)} \\
&= \sum_{j_a} P^{j_a}_{(j_1 t_1),(j_2 t_2),(j_3 t_3),(j_4 t_4)}\; C^{j_a}_{(j_1 m_1),(j_2 m_2),(j_3 m_3),(j_4 m_4)} \,.
\end{aligned} \tag{4.12}$$

While both t_a and m_a can be summed over, j_a has to stay as it is shared between the reduced tensors and the CGCs, giving both P and C an internal index. As an example,

suppose $j_1 = j_2 = j_3 = j_4 = \frac{1}{2}$ for a spin-$\frac{1}{2}$ system. Then $j_a = 0$ or $j_a = 1$, mediating either a spin-singlet or a spin-triplet between the pairs (j_1, j_2) and (j_3, j_4).

Fig. 4.3 illustrates the compactification of a rank-5 tensor, this time yielding two internal indices. Note that the direction of the external indices is not specified, but must be taken

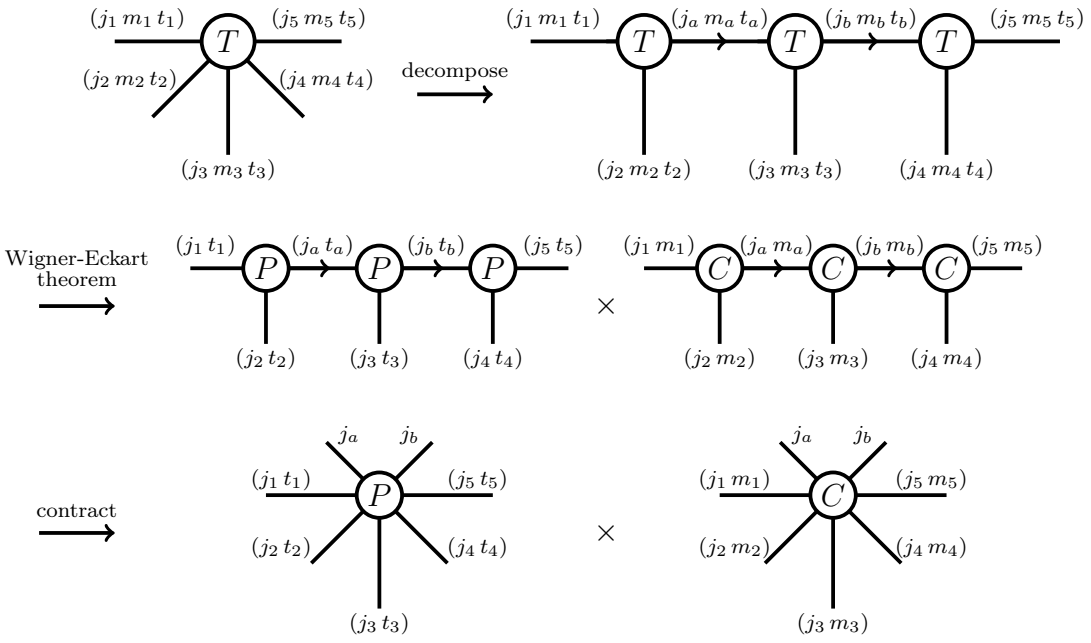


Figure 4.3: Compactification of an SU(2)-invariant rank-5 tensor.

into account if actual computations are performed. The direction of the internal indices is arbitrary and, without loss of generality, can be set as going from left to right. If this leads to a CGC with three ingoing or three outgoing indices, we simply attach a fourth bond with $l = m = 0$ in the opposite direction and insert a resolution of unity:

$$\langle j_1 m_1, j_2 m_2, j_3 m_3 | 00 \rangle = \sum_{j_4, m_4} \langle j_1 m_1, j_2 m_2 | j_4 m_4 \rangle \langle j_4 m_4, j_3 m_3 | 00 \rangle \tag{4.13}$$

$$= \langle j_1 m_1, j_2 m_2 | j_3 - m_3 \rangle \langle j_3 - m_3, j_3 m_3 | 00 \rangle \ . \tag{4.14}$$

We can now define SU(2)-invariant tensors of arbitrary rank r as

$$T_{(j_1 m_1 t_1),\ldots,(j_r m_r t_r)} = \sum_{j_{a_1},\ldots,j_{a_{r-3}}} P^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1 t_1),\ldots,(j_r t_r)} \, C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1 m_1),\ldots,(j_r m_r)} \, . \tag{4.15}$$

The partially contracted CGCs C are called *intertwiner* [99]. They have an important orthogonality relation,

$$\sum_{m_1,\ldots,m_r} C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1 m_1),\ldots,(j_r m_r)} \, C^{j_{a'_1},\ldots,j_{a'_{r-3}}}_{(j_1 m_1),\ldots,(j_r m_r)} = N^{j_{a_1},\ldots,j_{a_{r-3}}}_{j_1,\ldots,j_r} \, \delta_{j_{a_1}, j_{a'_1}}, \ldots, \delta_{j_{a_{r-3}}, j_{a'_{r-3}}} \, , \tag{4.16}$$

$$N^{j_{a_1},\ldots,j_{a_{r-3}}}_{j_1,\ldots,j_r} = \sum_{m_1,\ldots,m_r} C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1m_1),\ldots,(j_rm_r)} C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1m_1),\ldots,(j_rm_r)}\,, \tag{4.17}$$

which can be used to flesh out the reduced tensors P for a given full tensor T:

$$\begin{aligned}\sum_{j_{a'_1},\ldots,j_{a'_{r-3}}} P^{j_{a'_1},\ldots,j_{a'_{r-3}}}_{(j_1t_1),\ldots,(j_rt_r)} \sum_{m_1,\ldots,m_r} C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1m_1),\ldots,(j_rm_r)} C^{j_{a'_1},\ldots,j_{a'_{r-3}}}_{(j_1m_1),\ldots,(j_rm_r)}& \\ = \sum_{m_1,\ldots,m_r} C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1m_1),\ldots,(j_rm_r)}\, T_{(j_1m_1t_1),\ldots,(j_rm_rt_r)}&\end{aligned} \tag{4.18}$$

$$\begin{aligned}\sum_{j_{a'_1},\ldots,j_{a'_{r-3}}} P^{j_{a'_1},\ldots,j_{a'_{r-3}}}_{(j_1t_1),\ldots,(j_rt_r)}\, N^{j_{a_1},\ldots,j_{a_{r-3}}}_{j_1,\ldots,j_r}\, \delta_{j_{a_1},j_{a'_1}},\ldots,\delta_{j_{a_{r-3}},j_{a'_{r-3}}}& \\ = \sum_{m_1,\ldots,m_r} C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1m_1),\ldots,(j_rm_r)}\, T_{(j_1m_1t_1),\ldots,(j_rm_rt_r)}&\end{aligned} \tag{4.19}$$

$$\Rightarrow \quad P^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1t_1),\ldots,(j_rt_r)} = \left(N^{j_{a_1},\ldots,j_{a_{r-3}}}_{j_1,\ldots,j_r}\right)^{-1} \sum_{m_1,\ldots,m_r} C^{j_{a_1},\ldots,j_{a_{r-3}}}_{(j_1m_1),\ldots,(j_rm_r)}\, T_{(j_1m_1t_1),\ldots,(j_rm_rt_r)}\,. \tag{4.20}$$

In summary, the m-parameterization of SU(2)-invariant tensors is externalized into the intertwiners C, which can be operated on independent of the reduced tensors P.

Having motivated the structure of symmetric tensors, we now look at them from a computational perspective and discuss how they are actually stored. First, we form a *label* out of all the attached indices I_m, which serves as a global and unique identifier for each tensor within the network. Since an index is basically a parameterization of states in a Hilbert space, it has sets of quantum numbers associated with it, as was explained above. We call such a set a *channel* $c_{m,n}$, which can be, for example, the z-component of the spin (S_z) in the Heisenberg model or total spin and deviation from half-filling (j, c_z) in the Hubbard model. A set of channels $\{c_{m,n}\}$ is required to satisfy the abelian conservation laws in Eq. (4.9) and the SU(2)-selection rules given by the CGCs. Together with a degeneracy d_n, representing internal j-indices, it forms a *key* k_n, which can be used to access the corresponding dense tensor T_n. Table 4.1 shows the tabular structure. For

Table 4.1: Storage of a block-sparse tensor with indices I_m, channels $c_{m,n}$, degeneracies d_n, and dense tensors T_n.

I_1	I_2	...	I_r		
$c_{1,1}$	$c_{2,1}$	...	$c_{r,1}$	d_1	T_1
$c_{1,2}$	$c_{2,2}$	...	$c_{r,2}$	d_2	T_2
$c_{1,3}$	$c_{2,3}$	...	$c_{r,3}$	d_3	T_3
...					

sparse tensors that consist of a small number of large blocks, it suffices to store keys and corresponding dense tensors as pairs in a list. For a large number of small dense tensors, it may be necessary to use a memory pool to avoid overhead through fragmentation. Access of elements takes place either sequentially or associatively. For the former case, one should define an order for the keys and traverse according to it to avoid ambiguity. For the latter case, a hash table may be used.

The dense tensors themselves should be stored as one consecutive array in column-major order for compatibility with the Basic Linear Algebra Package (BLAS). For instance, the global indexing of a rank-5 tensor with indices $(i_1, i_2, i_3, i_4, i_5)$ and dimensions $(N_1, N_2, N_3, N_4, N_5)$ is

$$I = i_1 + i_2\, N_1 + i_3 N_2 N_1 + i_4 N_3 N_2 N_1 + i_5 N_4 N_3 N_2 N_1\,. \tag{4.21}$$

4.2 Permuting indices

As standard algebra routines only work on matrices, indices are expected to appear in a certain order. If they do not, they need to be permuted properly.

Consider the indexing of a dense tensor of rank r, which reads, schematically,

$$I_r = i_1 + i_2 N_1 + ... + i_{r-1} N_{r-2} N_{r-3} ... N_1 + i_r N_{r-1} N_{r-2} ... N_1\,. \tag{4.22}$$

Recursively, this gives

$$I_r = i_r \prod_{k=r-1}^{1} N_k + I_{r-1}; \quad I_1 = i_1\,. \tag{4.23}$$

A permutation is now defined as a new sequence of indices, for instance, $P\,(k) = 1, 4, 3, 2$ compared to the original order $k = 1, 2, 3, 4$ for a rank-4 tensor, as shown in Fig. 4.4.

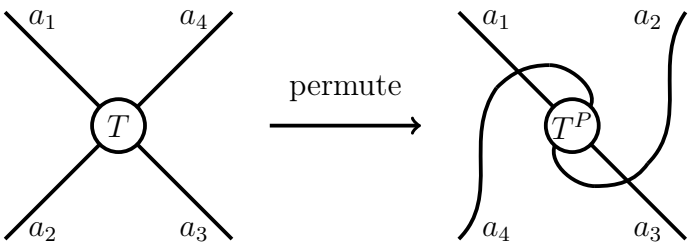


Figure 4.4: Permutation of a rank-4 tensor.

The new global index yields, in the general case,

$$I_r^P = i_{P(r)} \prod_{k=P^{-1}(r)-1}^{1} N_{P(k)} + I_{r-1}^P; \quad I_1^P = i_{P(1)} \prod_{k=P^{-1}(1)-1}^{1} N_{P(k)}\,, \tag{4.24}$$

where $P^{-1}\left(P\left(k\right)\right) = P\left(P^{-1}\left(k\right)\right) = k$. The reordered tensor T^P is then assigned through

$$T^P\left[I_r^P\right] = T\left[I_r\right] . \tag{4.25}$$

On the structural, SU(2)-symmetric level, the two different orders in Fig. 4.4 are represented as follows:

$$T_{(j_1m_1t_1),(j_2m_2t_2),(j_3m_3t_3),(j_4m_4t_4)} = \sum_{j_a} P^{j_a}_{(j_1t_1),(j_2t_2),(j_3t_3),(j_4t_4)}\; C^{j_a}_{(j_1m_1),(j_2m_2),(j_3m_3),(j_4m_4)} , \tag{4.26}$$

$$T_{(j_1m_1t_1),(j_4m_4t_4),(j_3m_3t_3),(j_2m_2t_2)} = \sum_{j'_a} P^{j'_a}_{(j_1t_1),(j_4t_4),(j_3t_3),(j_2t_2)}\; C^{j'_a}_{(j_1m_1),(j_4m_4),(j_3m_3),(j_2m_2)} . \tag{4.27}$$

For a given $P^{j_a}_{(j_1t_1),(j_2t_2),(j_3t_3),(j_4t_4)}$, one can thus multiply by the corresponding intertwiner, permute according to Eq. (4.24), and then isolate $P^{j'_a}_{(j_1t_1),(j_4t_4),(j_3t_3),(j_2t_2)}$ through Eq. (4.20):

$$P^{j'_a}_{(j_1t_1),(j_4t_4),(j_3t_3),(j_2t_2)} = \left(N^{j'_a}_{j_1,j_2,j_3,j_4}\right)^{-1} \sum_{j_a} P^{j_a}_{(j_1t_1),(j_2t_2),(j_3t_3),(j_4t_4)}\; X^{j_a,j'_a}_{(j_1,j_2,j_3,j_4),(j_1,j_4,j_3,j_2)} , \tag{4.28}$$

$$X^{j_a,j'_a}_{(j_1,j_2,j_3,j_4),(j_1,j_4,j_3,j_2)} = \sum_{m_1,m_2,m_3,m_4} C^{j_a}_{(j_1m_1),(j_2m_2),(j_3m_3),(j_4m_4)}\; C^{j'_a}_{(j_1m_1),(j_4m_4),(j_3m_3),(j_2m_2)} . \tag{4.29}$$

To every single j_a, there are potentially multiple corresponding j'_a, which is why the overlap $X^{j_a,j'_a}_{\dots}$ has to be calculated for all possible values of j'_a. The generalization to arbitrary permutations should be self-explaining.

4.3 Pairwise contraction

One of the elementary operations is the pairwise multiplication or *contraction* of tensors. Consider T^1 with rank r_1 and T^2 with r_2 and let l be the number of indices they have in common. Once both tensors are permuted with their common indices in the same order and next to each other, their contraction can then be written as

$$T^1_{a_1\dots a_k b_1\dots b_l} T^2_{b_1\dots b_l c_1\dots c_m} = T^3_{a_1\dots a_k c_1\dots c_m} , \tag{4.30}$$

with $r_1 = k + l$, $r_2 = l + m$, and the rank of the the resulting tensor $r_3 = k + m$. Fig. 4.5 gives a graphical depiction of such a contraction. The contraction of each common index b_i has to be viewed as $\sum_m \langle\psi_m|\psi_m\rangle$, where m enumerates the states within the Hilbert space b_i represents.

In practice, contraction is carried out on two levels. On the top, structural, level, we temporarily create a map from one *sub-key*, which stores the channels of the subset

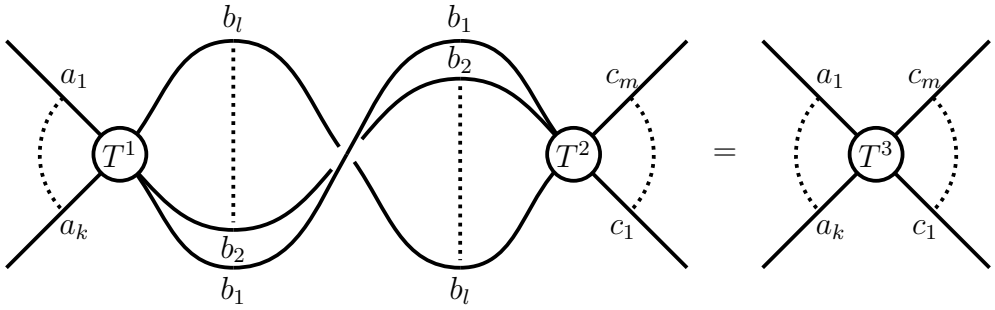


Figure 4.5: Pairwise tensor contraction

common indices, to multiple full keys that share this sub-key, and their dense tensors. For one sub-key, all corresponding dense tensors of T_1 and T_2 are then contracted with each other. The key of the results is made out of the channels of indices that T_1 and T_2 do not have in common. If, for example, we contract a rank-5 tensor with a rank-4 tensor that have two indices in common, their labels may be $(I_a, I_b, I_c, I_d, I_e)$ and (I_d, I_e, I_f, I_g). The sub-keys are then all unique sets of channels belonging to (I_d, I_e) and the resulting label is $(I_a, I_b, I_c, I_f, I_g)$.

Using SU(2)-symmetries requires an additional layer of processing. Suppose one is given two rank-4 tensors with two common indices. Analogously to the procedure in Sec. 4.2, we think of the reduced tensors as being multiplied by their intertwiners, contracted as usual, and then stripped of the m-dependency through Eq. (4.20):

$$
\begin{aligned}
&T_{(j_1 m_1 t_1),(j_2 m_2 t_2),(j_5 m_5 t_5),(j_6 m_6 t_6)} \\
&\quad = \sum_{j_3,m_3,t_3} \sum_{j_4,m_4,t_4} T_{(j_1 m_1 t_1),(j_2 m_2 t_2),(j_3 m_3 t_3),(j_4 m_4 t_4)}\, T_{(j_3 m_3 t_3),(j_4 m_4 t_4),(j_5 m_5 t_5),(j_6 m_6 t_6)}\,, \\
&\Rightarrow \quad \sum_{j_c} P^{j_c}_{(j_1 t_1),(j_2 t_2),(j_5 t_5),(j_6 t_6)}\; C^{j_c}_{(j_1 m_1),(j_2 m_2),(j_5 m_5),(j_6 m_6)} \\
&\qquad = \sum_{j_3,m_3,t_3} \sum_{j_4,m_4,t_4} \sum_{j_a} P^{j_a}_{(j_1 t_1),(j_2 t_2),(j_3 t_3),(j_4 t_4)}\; C^{j_a}_{(j_1 m_1),(j_2 m_2),(j_3 m_3),(j_4 m_4)} \\
&\qquad\quad \cdot \sum_{j_b} P^{j_b}_{(j_3 t_3),(j_4 t_4),(j_5 t_5),(j_6 t_6)}\; C^{j_b}_{(j_3 m_3),(j_4 m_4),(j_5 m_5),(j_6 m_6)}\,, \\
&\Rightarrow \quad P^{j_c}_{(j_1 t_1),(j_2 t_2),(j_5 t_5),(j_6 t_6)} \\
&\qquad = \left(N^{j_c}_{j_1,j_2,j_5,j_6}\right)^{-1} \sum_{j_a,j_b} P^{j_a}_{(j_1 t_1),(j_2 t_2),(j_3 t_3),(j_4 t_4)}\; P^{j_b}_{(j_3 t_3),(j_4 t_4),(j_5 t_5),(j_6 t_6)}
\end{aligned}
$$

$$\cdot\, Y^{j_a,j_b,j_c}_{(j_1,j_2,j_3,j_4),(j_3,j_4,j_5,j_6),(j_1,j_2,j_5,j_6)}\,, \tag{4.31}$$

$$\begin{aligned} &Y^{j_a,j_b,j_c}_{(j_1,j_2,j_3,j_4),(j_3,j_4,j_5,j_6),(j_1,j_2,j_5,j_6)} \\ &\quad= \sum_{m_1,\ldots,m_6} C^{j_a}_{(j_1 m_1),(j_2 m_2),(j_3 m_3),(j_4 m_4)}\, C^{j_b}_{(j_3 m_3),(j_4 m_4),(j_5 m_5),(j_6 m_6)} \\ &\qquad\cdot\, C^{j_c}_{(j_1 m_1),(j_2 m_2),(j_5 m_5),(j_6 m_6)}\,. \end{aligned} \tag{4.32}$$

The calculation of $Y^{j_a,j_b,j_c}_{\ldots}$ can be significantly expensive if not implemented properly. Hence, the intertwiners are best left fragmented as their individual CGCs, which are then contracted within Eq. (4.32) in the order which keeps the rank of intermediate results as low as possible. For the generalization to arbitrary contractions, one needs to keep in mind that, instead of individual, internal indices, j_a, j_b, and j_c, one needs to operate on tupels $\{j_a\}$, $\{j_b\}$, and $\{j_c\}$, for ranks higher than 4. Otherwise, the scheme is structurally identical to the special case above.

On the lower, degenerate level, dense tensors are contracted by an effective matrix-matrix multiplication where all common and distinct indices are bundled into one multi-index. In the most general case, i.e., of Eq. (4.30), this translates to

$$T^1\left[A + B\cdot N_A\right] \times T^2\left[B + C\cdot N_B\right] = T^3\left[A + C\cdot N_A\right]\,, \tag{4.33}$$

where

$$A = a_1 + a_2\, N_{a_1} + \ldots + a_k\, N_{a_{k-1}}\ldots N_{a_1}, \qquad N_A = N_{a_{k-1}}\ldots N_{a_1}\,, \tag{4.34}$$

$$B = b_1 + b_2\, N_{b_1} + \ldots + b_l\, N_{b_{l-1}}\ldots N_{b_1}, \qquad N_B = N_{b_{l-1}}\ldots N_{b_1}\,, \tag{4.35}$$

$$C = c_1 + c_2\, N_{c_1} + \ldots + c_m\, N_{c_{m-1}}\ldots N_{c_1}, \qquad N_C = N_{c_{m-1}}\ldots N_{c_1}\,. \tag{4.36}$$

4.4 Fusion and splitting

In order to decompose tensors, multiple indices need to be *fused* together and *split* afterwards.

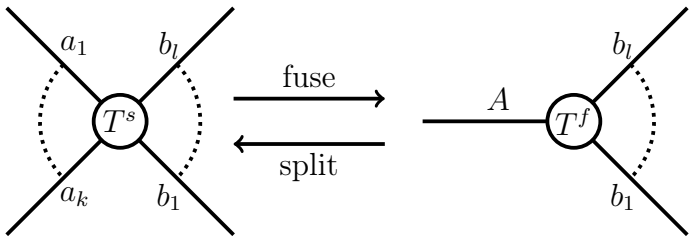


Figure 4.6: Fusion and splitting.

For both U(1) and SU(2) symmetries, fusion takes place on the structural, as well as the degenerate level. As an introduction, suppose a tensor has two bonds $|m_1\, t_1\rangle$ and $|m_2\, t_2\rangle$ with abelian quantum numbers m_i, orders of degeneracy d_i, and dense indices $t_i = 1, ..., d_i$. Both are merged by a projection onto one Hilbert space with states $|m\, t\rangle$,

$$|m\, t\rangle = \sum_{m_1,t_1,m_2,t_2} |m_1\, t_1, m_2\, t_2\rangle \, \langle m_1\, t_1, m_2\, t_2 | m\, t\rangle \tag{4.37}$$

$$\langle m_1 t_1, m_2 t_2 | m t\rangle = \delta_{m_1+m_2,m}\, \delta_{t_1+t_2\cdot d_1,t}\,, \tag{4.38}$$

where the second line follows from the conservation law in Eq. (4.7) and the orthogonality of degenerate states. The dimension of the final dense index t is given by

$$d = \sum_{i_1,i_2} d_{i_1}\, d_{i_2}\, \delta_{m_{i_1}+m_{i_2},m}\,. \tag{4.39}$$

If the projection matrix $\langle m_1 t_1, m_2 t_2 | m t\rangle$ is arranged so that $\langle m_1\, t_1, m_2\, t_2|$ and $|m\, t\rangle$ parameterize rows and columns, respectively, the individual dense blocks for varying m_1 and m_2 and equal m look, schematically, as follows:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.40}$$

One can now easily generalize this scheme picture to the case in Fig. 4.6,

$$Q = \langle m_{a_1}\, t_{a_1}, ..., m_{a_k}\, t_{a_k} | m_A\, t_A\rangle = \delta_{s_{a_1} m_{a_1}+,...,+s_{a_k} m_{a_k}, s_A m_A}\, \delta_{t_{a_1}+t_{a_2}\cdot d_{a_1}+...+t_{a_k}\cdot d_{a_1}\cdot....\cdot d_{a_{k-1}}}\,, \tag{4.41}$$

where the s_i indicate the directions of bonds. Singh refers to Q as the fuse table [97], whereas the Itensor-library calls it the combiner [100]. The bottom line is that fusion is the operation through which new quantum numbers m_A are generated. The splitting of a joint index is obviously performed by applying $Q^\dagger$.

For a given SU(2)-invariant tensor, we consider again the rank-5 tensor from Fig. 4.3 as an example and conceptualize its reduced version as a sequence of rank-three tensors with internal, angular momenta j_a and j_b and fictitious, internal, dense indices t_a and t_b. Suppose that the indices 3, 4, and 5 have to be fused, as depicted in Fig. 4.7. Due to the selection rules of CGCs, the external angular momenta, j_3, j_4, and j_5, do not uniquely identify the outcome of fusion, in contrast to the U(1)-case discussed above. Instead, the final index of the projector is dictated by the internal index of the reduced tensor, in this case j_a. To isolate it, we think of the reduced projector as a sequence of rank-3 tensors

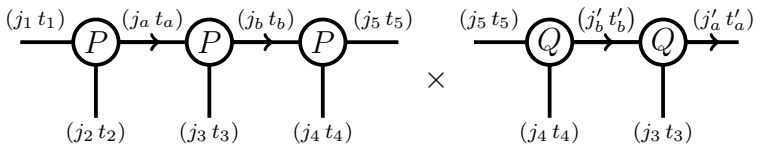


Figure 4.7: Fusion of three external indices in an SU(2)-invariant rank-5 tensor.

as well, shown as two Q's with the reverse order of indices in Fig. 4.7. The contraction of the two adjacent P and Q yields a result proportionate to $\delta_{j_b,j'_b}\,\delta_{t_b,t'_b}$ due to SU(2)-invariance, which can be absorbed to either the left or the right. Afterwards, the middle P is contracted with the remaining Q along $(j_b t_b)$ and $(j_3 t_3)$ to finally expose $(j_a t_a)$. The dimension d_a of t_a is determined analogously to Eq. (4.39):

$$d_a = \sum_{j_3,j_4,j_5,j_b} (d_3\,d_4\,d_5)_{j_3,j_4,j_5,j_b}\ . \tag{4.42}$$

The next step is to determine the proper prefactor $Q^{j'_b}_{j_5\,j_4\,j_3\,j'_a}$, which captures the SU(2)-symmetry and is used instead of a bare 1 as in Eq. (4.40). With this in mind, we note that the reduced projector $Q^{j'_b}_{(j_5 t_5),(j_4 t_4),(j_3 t_3),(j'_a t'_a)}$ is accompanied by the intertwiner $C^{j'_b}_{(j_5 m_5),(j_4 m_4),(j_3 m_3),(j'_a m'_a)}$. To ensure $Q\,Q^\dagger = \mathbb{1}$, the following condition must then hold

$$\begin{aligned} Q^{j'_b}_{j_5\,j_4\,j_3\,j'_a} Q^{j'_b}_{j_5\,j_4\,j_3\,j''_a} \sum_{m_5,m_4,m_3} C^{j'_b}_{(j_5 m_5),(j_4 m_4),(j_3 m_3),(j'_a m'_a)}\, C^{j'_b}_{(j_5 m_5),(j_4 m_4),(j_3 m_3),(j''_a m''_a)} \\ = \delta_{j'_a,j''_a}\,\delta_{m'_a,m''_a}\,, \end{aligned} \tag{4.43}$$

$$\Rightarrow \quad Q^{j'_b}_{j_5\,j_4\,j_3\,j'_a} = \left(\sum_{m_5,m_4,m_3} C^{j'_b}_{(j_5 m_5),(j_4 m_4),(j_3 m_3),(j'_a m'_a)}\, C^{j'_b}_{(j_5 m_5),(j_4 m_4),(j_3 m_3),(j'_a m'_a)} \right)^{-\frac{1}{2}}\,, \tag{4.44}$$

with the right hand side of Eq. (4.44) being independent of m'_a due to SU(2)-invariance. The proper prefactor is thus the final ingredient for the construction of SU(2)-invariant projectors. As in the case of permutation and contraction, we assume that the example is sufficient to explain the general case.

4.5 Decomposition

Consider an arbitrary real or complex matrix M with m rows and n columns. The singular value decompostion (SVD) performs the following factorization

$$M = U\,\Sigma\,V^T \tag{4.45}$$

with U and V^T unitary $m \times d$ and $d \times n$ matrices respectively, whereas $d = \min(m, n)$. Here Σ is a diagonal $d \times d$ matrix and can be written as diag $(\sigma_1, ..., \sigma_d)$ with the *singular values* σ_i. For symmetric M, the SVD becomes the eigenvalue decomposition with eigenvectors $U = V$ and eigenvalues σ_i.

In tensor networks, the SVD is used to factorize one tensor into two and generate a virtual bond between them. Preceeded by fusion and succeeded by splitting, the operation, depicted in Fig. 4.8, can be performed on all dense tensors of T_f individually using stan-

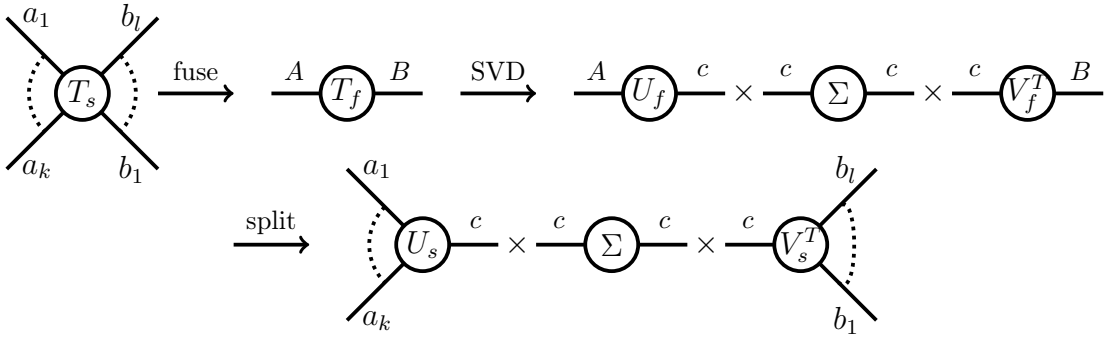


Figure 4.8: Decomposition of tensor $T_{a_1,...,a_k,b_1,...,b_l}$.

dard routines. The singular values σ_i are the weights of the states of the newly created Hilbert space, parameterized by c. If T_s is part of an acyclic tensor-network state, they are equal to the square roots of the eigenvalues of the reduced density matrix of a mixed state and can be used to calculate the von-Neumann entropy

$$S = -\sum_i \sigma_i^2 \ln {\sigma_i}^2 \,, \tag{4.46}$$

which quantifies to what extent the two subsystems that c connects are entangled.

If an SVD is carried out after an optimization process that explores new subsectors of the total Hilbert space, the dimension N_c can be as large as min $(N_{a_1} \cdot ... \cdot N_{a_2}, N_{b_1} \cdot ... \cdot N_{b_l})$, which is why it is usually done in conjuction with a *truncation*, in which only the m largest singular values are chosen, with m being the *maximum bond dimension* or, equivalently, the *maximum number of states*. All other values are dropped, and the corresponding columns of U and rows of V^T are erased, which prevents the tensor network from growing exponentially. For degenerate states, multiple singular values can be equal, in which case the truncation needs to be slightly modified by either picking or dropping the whole block of degenerate σ_i, according to whether or not they cross the threshold set by m. Otherwise, one breaks up multiplets, which must be done arbitrarily and thus leads to an indeterministic algorithm.

Subsequently, one needs to decide whether to multiply one of U_S or V_s^T with Σ, or if it should be split into $\sqrt{\Sigma} \cdot \sqrt{\Sigma}$ and factored into both sides [101]. What option to take depends on the context and will be discussed subsequently. In the last case, we describe

how we can additionally improve numerical stability by redistributing weight. Suppose that, for example, U and V^T were updated as $U \cdot \sqrt{\Sigma}$ and $\sqrt{\Sigma} \cdot V^T$; the distribution of singular values is usually exponential due to the area law, which can lead to a sharp gradient of weight within the tensors and ultimately to rounding errors that accumulate over time. To circumvent this, we examine each quantum number sector of c individually and insert multiple Givens rotations G_i between the two $\sqrt{\Sigma}$ to smear out the weight evenly between the columns of U and rows of V^T:

$$U\sqrt{\Sigma} \cdot \sqrt{\Sigma}\, V^T = U\sqrt{\Sigma}\, G_1\, G_2 ... G_n \cdot G_n^\dagger ... G_2^\dagger\, G_1^\dagger\, \sqrt{\Sigma}\, V^T\,. \tag{4.47}$$

Since Givens rotations are unitary, the tensor network as a whole is not changed. If, for example, a channel consists of 4 states, the first rotation applied to the corresponding dense tensor of U will be redistributed between the first and the last column, because σ_i are originally stored in descending order, and we want to take care of the largest disparity first:

$$U\sqrt{\Sigma} \cdot G_1 = \begin{pmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ u_{21} & u_{22} & u_{23} & u_{24} \\ ... & ... & ... & ... \end{pmatrix} \cdot \mathrm{diag}\,(\sigma_1', \sigma_2', \sigma_3', \sigma_4') \cdot \begin{pmatrix} c & 0 & 0 & s \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s & 0 & 0 & c \end{pmatrix}, \tag{4.48}$$

$$= \begin{pmatrix} u_{11}\sigma_1' c - u_{14}\sigma_4' s & u_{12}\sigma_2' & u_{13}\sigma_3' & u_{11}\sigma_1' s + u_{14}\sigma_4' c \\ u_{21}\sigma_1' c - u_{24}\sigma_4' s & u_{22}\sigma_2' & u_{23}\sigma_3' & u_{21}\sigma_1' s + u_{24}\sigma_4' c \\ ... & ... & ... & ... \end{pmatrix}, \tag{4.49}$$

with $c = \cos\phi$, $s = \sin\phi$, $t = \tan\phi$ and $\sigma_i' = \sqrt{\sigma_i}$. We now set

$$c\sigma_1' - s\sigma_4' = s\sigma_1' + c\sigma_4' \tag{4.50}$$

which gives

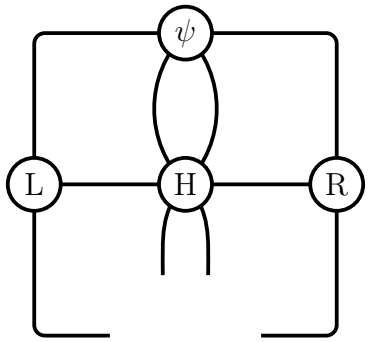
$$t = \frac{\sigma_1' - \sigma_4'}{\sigma_1' + \sigma_4'}, \quad s = \frac{t}{\sqrt{1+t^2}}, \quad c = \frac{1}{\sqrt{1+t^2}}\,. \tag{4.51}$$

After the rotation is performed, the weights corresponding to columns 1 and 4 are reset to $\sigma_1' \to c\sigma_1' - s\sigma_4'$ and $\sigma_4' \to s\sigma_1' + c\sigma_4'$. Now we look for the pair with largest difference in the updated sequence of $(\sigma_1', \sigma_2', \sigma_3', \sigma_4')$ and perform a Givens rotation for these columns. This process is repeated until one iteratively arrives at an equal distribution of weight among the columns. The rows of V^T are processed equivalently.

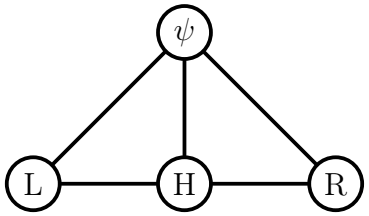
4.6 Graph-guided contraction

One of the main bottlenecks in the variational optimization of any tensor network state is the contraction of multiple block-sparse tensors, which takes place either in the preparation of the environment of an effective Hamiltonian, or in the iterative diagonalization of a part of the wave function. If the tensors are contracted one after another and, as a whole, as described in Sec. 4.3, the number of elements of the intermediate results can be several orders of magnitude larger than all of the inital tensors taken together, which poses a strong limitation on the number of states one can keep. To circumvent this problem, one could instead loop over the dense elements of the first tensor, perform the whole contraction sequence for each of them, and, then add up all of the results. This keeps the memory usage below a reasonable threshold, but increases the calculation time significantly, as identical intermediate contractions need to be performed multiple times. In the following, we present an algorithm that combines the best of both worlds, namely, minimal peak-memory usage combined with maximum speed.

As an example, we consider the two-site optimization step in the density matrix renormalization group, which is depicted in Fig. 4.9(a). Here ψ is the part of the wave function



(a) Illustration of the operation $H_{eff}\,|\psi\rangle$ within the DMRG, where $H_{eff} = L \cdot H \cdot R$.



(b) $H_{eff}\,|\psi\rangle$ reinterpreted as a graph. L, R, H and ψ are supervertices which consist of multiple bare vertices.

Figure 4.9: Translation of a tensor network to a graph.

we want to optimize, H is the corresponding portion of the Hamiltonian, L and R are partial contractions of the overall tensor network, and $L \cdot H \cdot R$ forms the effective Hamiltonian of the reduced Hilbert space. The details of exactly how this arrangement arises will be discussed in Sec. 5.4.

The first step is to interpret a sparse tensor as a supervertex and each of its pairs of a key and a dense tensor as a *vertex*, which transforms the tensor network into a hypergraph, in which multiple vertices of adjacent supervertices are connected by a common set of sub-keys, as shown in Fig. 4.9(b). We associate each label with a spectral *color*, which is implemented as a bit field with one bit set to one and all others set to zero. The color c_i of a contraction result is determined taking the logical or of the two initial colors, i.e., $c_{A\cdot B} = c_A \vee c_B$. The final contraction result contains all of the spectral components and

can hence be considered as "white". The number of spectral components within a color is called its density.

For the case of Fig. 4.9(b), the colors for the contraction sequence $\psi \to L \to H \to R$ look like:

$$c_\psi = (1,0,0,0)\,, \quad c_L = (0,1,0,0)\,, \quad c_R = (0,0,1,0)\,, \quad c_H = (0,0,0,1)\ , \tag{4.52}$$

$$c_{\psi\cdot L} = (1,1,0,0)\,, \quad c_{\psi\cdot L\cdot H} = (1,1,0,1)\,, \quad c_{\psi'=\psi\cdot L\cdot H\cdot R} = (1,1,1,1)\ . \tag{4.53}$$

As the example shows, the size of the bit field is determined by the number of spectral components, which is equal to the number of inital sparse tensors. Furthermore, two colors about to be contracted must at all times be distinct, meaning that

$$c_i \wedge c_j \stackrel{!}{=} (0,0,...,0)\ . \tag{4.54}$$

The color c is now attributed to every single vertex v, which yields its complete definition as the tupel (c,k,t). The contraction of two vertices v_i and v_j is defined as

$$(c_i,k_i,t_i) \times (c_j,k_j,t_j) = (c_i \vee c_j, k_i \triangle k_j, t_i \times t_j)\ , \tag{4.55}$$

and is illustrated in Fig. 4.10. Here $k_i \triangle k_j$ is the symmetric difference of both keys with respect to their labels, while $t_i \times t_j$ denotes the actual contraction of two dense tensors, which is performed by a proper reordering of indices, as shown in Eq. (4.30), and a subsequent matrix-matrix-multiplication. For SU(2) - symmetries, Eq. (4.31) must also be taken into account.

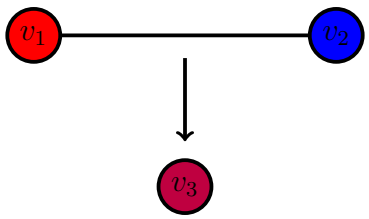


Figure 4.10: Vertex contraction.

Vertices of two adjacent supervertices with colors c_i and c_j are connected by a *hyperedge* E, which is defined by the following tuple:

$$E = (c_i, c_j, s_\alpha, \{v_{i,s_\alpha}\}, \{v_{j,s_\alpha}\})\ . \tag{4.56}$$

Here s_α is a sub-key consisting of multiple channels with its size determined by the number of indices the two tensors have in common. Contracting the edge consists of contracting every v_{i,s_α} with every v_{j,s_α}, as shown in Fig. 4.11. The hypergraph can now be initialized by traversing all sparse tensors, forming vertices out of their elements, and connecting

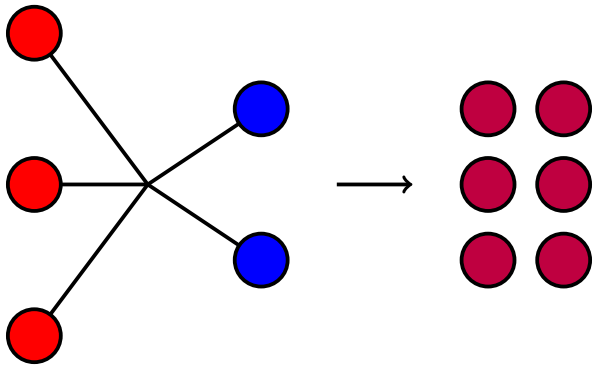


Figure 4.11: Hyperedge contraction.

them via hyperedges.

The idea of graph-guided contraction is to always contract the edge whose resulting color has the highest density, which leads to a recursive process that contracts first those vertices that have been created last. Consider, for example, a sequence of three sparse matrices that have to be multiplied one after another, as shown in Fig. 4.12. They are assigned the colors $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$, illustrated as red, blue, and green. For a pairwise contraction, one would first contract all red with all blue vertices. Here, however, we only contract one red-blue edge and then immediately create new edges connecting purple to green, whose sub-keys are inherited by the original blue-green edges. The new edges are now contracted before the second red-blue edge. Note how this implies that, at all times, at least one color of an edge has a density of one, meaning that it is always attached to at least one inital sparse tensor.

While this scheme has no significant impact on matrix multiplication, which we have only used as an example to convey the idea, the advantage for higher-rank tensors is huge, as the rank, and hence the size, of intermediate results usually exceeds those of the initial and the final tensor. We therefore only need to store a fraction of the large, temporary, sparse tensors, and then immediately contract them with the next vertices in line and erase them afterwards.

The order of contraction on the sparse level depends on the individual case and is chosen so that the largest rank that occurs intermittently, which will define the overall scaling, is minimal.

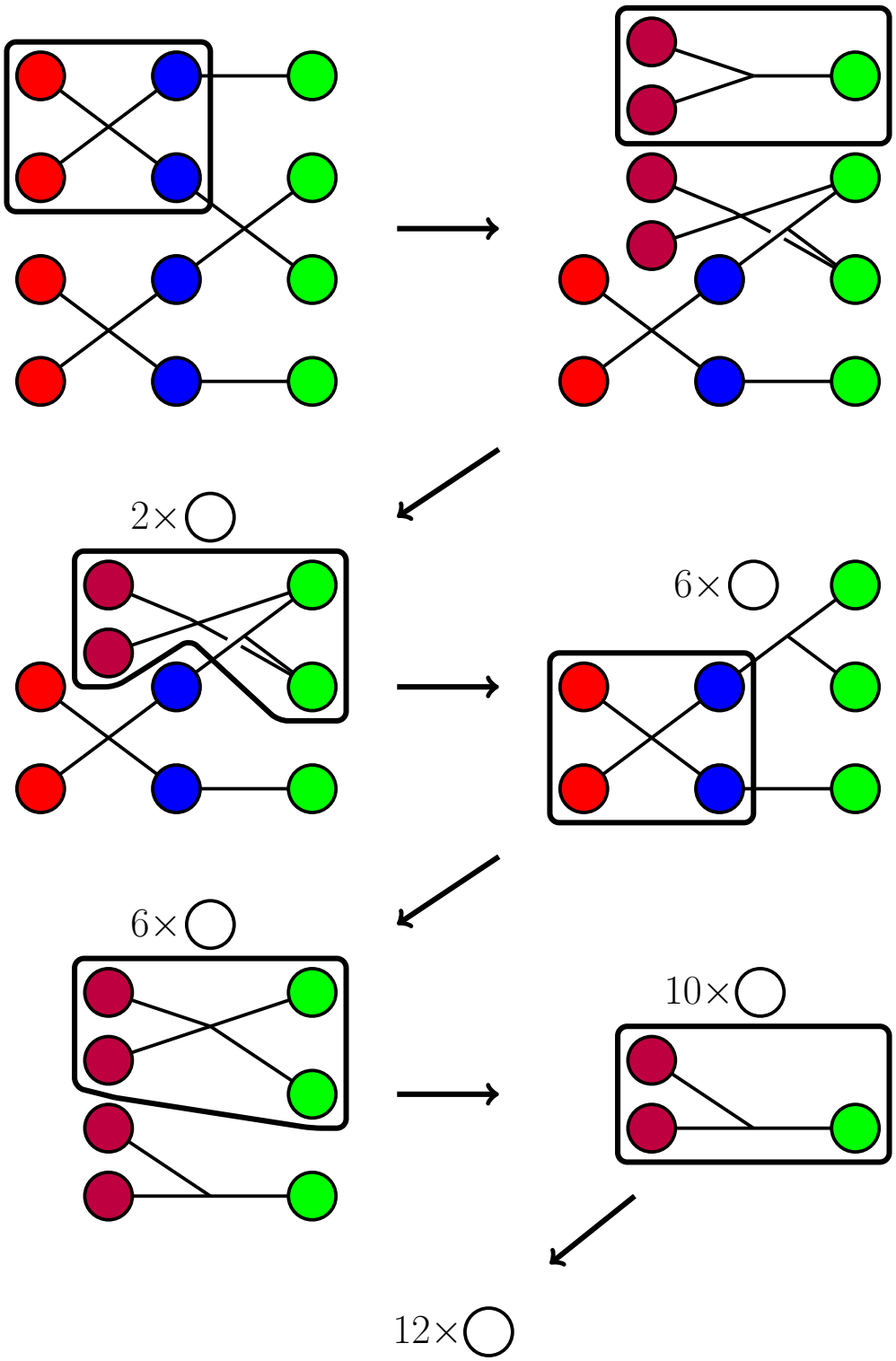


Figure 4.12: Graph-guided contraction for a sequence of three sparse matrices. The boxes denote what hyperedge is about to be contracted next, while the white vertices with no connections are part of the final sparse tensor.

5 One-dimensional systems

5.1 Matrix product states

The simplest case of a tensor network state is a *matrix product state* (MPS), which can be seen as the generalization of the AKLT-states described in Sec. 3.1 for arbitrary, one-dimensional systems. Consider a generic many-body wavefunction on a chain of N sites

$$|\psi\rangle = \sum_{j_1,...,j_N} \psi_{j_1,...,j_N} |j_1, ..., j_N\rangle \ . \tag{5.1}$$

Here $|j_1, ..., j_N\rangle = |j_1\rangle \otimes ... \otimes |j_N\rangle$ are many-body basis states with $j_i = 1, ..., d$, where d is the dimension of a local space, e.g., $d = 2$ for spin-1/2-systems or $d = 4$ for fermions in the Hubbard-model. The $\psi_{j_1,...,j_N}$ are coefficients and can be consolidated as a rank-N tensor, which, in general, grows exponentially with N. In the classical limit, they decompose as $\psi_{j_1,...,j_N} = \psi_{j_1} \cdot \psi_{j_2} \cdot ... \cdot \psi_{j_N}$.

We now apply the Schmidt decomposition to $|\psi\rangle$, which states that any state in $\mathcal{H} = \mathcal{H}_L \otimes \mathcal{H}_R$ can be written as

$$|\psi\rangle = \sum_\alpha \Lambda_\alpha |\alpha_L\rangle \otimes |\alpha_R\rangle \,, \quad \sum_\alpha \Lambda_\alpha^2 = 1 \,, \tag{5.2}$$

where we have split the chain of N sites into a left and right system $L = [1, ..., i]$ and $R = [i + 1, ..., N]$. Here $|\alpha_L\rangle$ and $|\alpha_R\rangle$ form orthogonal bases for L and R, respectively, which are connected by the non-negative Schmidt-coefficients Λ_α. For strong entanglement, many Λ_α will have similar positive values. For no entanglement at all, precisely one Schmidt coefficient would be one, with all others zero, which means that the two wavefunctions of L and R can be combined through a product state.

The Schmidt decomposition can be carried out by forming the reduced density matrix that represents a mixed state, which is done by tracing out one of the subsystems, such as L

$$\rho_R = \mathrm{Tr}_L \left(|\psi\rangle \langle\psi| \right) = \sum_{j_1,...,j_i} \langle j_1, ..., j_i|\psi\rangle \langle\psi|j_1, ..., j_i\rangle \tag{5.3}$$

$$= \sum_{\substack{j_{i+1},...,j_N \\ j'_{i+1},...,j'_N}} \rho_{\substack{j_{i+1},...,j_N \\ j'_{i+1},...,j'_N}} |j_{i+1}, ..., j_N\rangle \left\langle j'_{i+1}, ..., j'_N \right| \ , \tag{5.4}$$

reshaping the coefficients of ρ_R into a matrix $\rho_{j,j'}$, and calculating the eigenvalues and eigenvectors, which are given by Λ_α^2 and $|\alpha_R\rangle$, respectively. The $|\alpha_L\rangle$ are then simply given by $\langle\alpha_R|\psi\rangle / \Lambda_\alpha$. Alternatively, one could reshape $\psi_{j_1,...,j_N}$ itself into a matrix, $\psi_{j,j'}$,

with $j = (j_1, ..., j_i)$ and $j = (j_{i+1}, ..., j_N)$, and apply an SVD

$$\psi_{j,j'} = U_{j,\alpha} \Sigma_{\alpha,\alpha} V^T_{\alpha,j'} \,. \tag{5.5}$$

Here $U_{j,\alpha}$ and $V^T_{\alpha,j'}$ correspond to $|\alpha_L\rangle$ and $|\alpha_R\rangle$, respectively, while the singular values $\Sigma_{\alpha,\alpha}$ are equal to the Schmidt coefficients Λ_α. The SVD is preferable as it requires fewer operations.

Having explained how to execute the Schmidt-decomposition in practice, we apply it both between $[1, ..., i-1]$ and $[i, ..., N]$, and between $[1, ..., i]$ and $[i+1, ..., N]$:

$$|\psi\rangle = \sum_{\alpha_{i-1}} \Lambda_{\alpha_{i-1}} |\alpha_{i-1}\rangle_{[1,...,i-1]} |\alpha_{i-1}\rangle_{[i,...,N]} \tag{5.6}$$

$$|\psi\rangle = \sum_{\alpha_i} \Lambda_{\alpha_i} |\alpha_i\rangle_{[1,...,i]} |\alpha_i\rangle_{[i+1,...,N]} \,. \tag{5.7}$$

Two resolutions of unit are now inserted into Eq. (5.6), one in the basis of $|\alpha_i\rangle_{[i+1,...,N]}$ from Eq. (5.7) and one for the local basis on site i

$$|\psi\rangle = \sum_{j_i,\alpha_{i-1},\alpha_i} \Lambda_{\alpha_{i-1}} |\alpha_{i-1}\rangle_{[1,...,i-1]} |j_i\rangle \langle j_i|\alpha_i\rangle_{[i+1,...,N]} \langle\alpha_i|_{[i+1,...,N]} |\alpha_{i-1}\rangle_{[i,...,N]} \tag{5.8}$$

$$= \sum_{j_i,\alpha_{i-1},\alpha_i} |\alpha_{i-1}\rangle_{[1,...,i-1]} \Lambda_{\alpha_{i-1}} \langle j_i| \langle\alpha_i|_{[i+1,...,N]} |\alpha_{i-1}\rangle_{[i,...,N]} |j_i\rangle |\alpha_i\rangle_{[i+1,...,N]} \,. \tag{5.9}$$

We define the rank-3 tensor

$$\Gamma^{j_i}_{\alpha_{i-1},\alpha_i} = \langle j_i| \langle\alpha_i|_{[i+1,...,N]} |\alpha_{i-1}\rangle_{[i,...,N]} \,, \tag{5.10}$$

which is stored as described in Table 4.1. Inserting Eq. (5.10) into Eq. (5.9) yields

$$|\psi\rangle = \sum_{j_i,\alpha_{i-1},\alpha_i} |\alpha_{i-1}\rangle_{[1,...,i-1]} \Lambda_{\alpha_{i-1}} \Gamma^{j_i}_{\alpha_{i-1},\alpha_i} |j_i\rangle |\alpha_i\rangle_{[i+1,...,N]} \,. \tag{5.11}$$

Site i was thus separated from the wave function through a tensor that maps two virtual spaces to a physical one. If we repeat this procedure for all remaining sites, we arrive at a matrix product state for open boundary conditions, expressed in Vidal's notation [102],

$$|\psi\rangle = \sum_{\substack{j_1,...,i_N \\ \alpha_1,...,\alpha_{N-1}}} \Gamma^{j_1}_{\alpha_1} \Lambda_{\alpha_1} \Gamma^{j_2}_{\alpha_1,\alpha_2} \Lambda_{\alpha_2} \Gamma^{j_3}_{\alpha_2,\alpha_3} \Lambda_{\alpha_3} ... \Lambda_{\alpha_{N-1}} \Gamma^{j_N}_{\alpha_{N-1}} |j_1, ..., j_N\rangle \,, \tag{5.12}$$

which is illustrated in Fig. 5.1. Such an MPS can be seen as a simple example of a tree-tensor network shown in Fig. 4.2(a). Here the j_i are physical bonds used to measure obervables, while the α_i are virtual bonds which encode entanglement.

Usually, the Schmidt coefficients Λ_{α_i} are not stored separately, but are attached to the

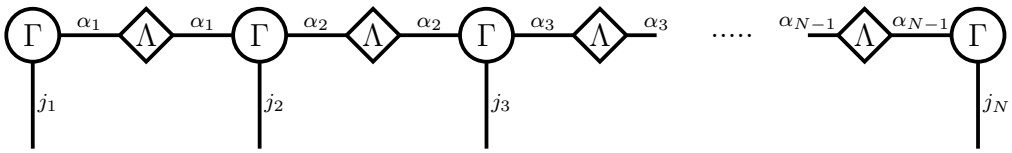


Figure 5.1: Matrix product state in Vidal's notation.

right or to the left tensor, which leads to left canonical

$$\Gamma^{j_1}_{\alpha_1} = A^{j_1}_{\alpha_1}, \quad \Lambda_{\alpha_{i-1}}\Gamma^{j_i}_{\alpha_{i-1},\alpha_i} = A^{j_i}_{\alpha_{i-1},\alpha_i}, \quad \Lambda_{\alpha_{N-1}}\Gamma^{j_N}_{\alpha_{N-1}} = A^{j_N}_{\alpha_{N-1}} \tag{5.13}$$

or right-canonical

$$\Gamma^{j_1}_{\alpha_1}\Lambda_{\alpha_1} = B^{j_1}_{\alpha_1}, \quad \Gamma^{j_i}_{\alpha_{i-1},\alpha_i}\Lambda_{\alpha_i} = B^{j_i}_{\alpha_{i-1},\alpha_i}, \quad \Gamma^{j_N}_{\alpha_{N-1}} = B^{j_N}_{\alpha_{N-1}} \tag{5.14}$$

MPSs. The tensors A and B have the following normalization properties:

$$\sum_{j_i}\sum_{\alpha_{i-1}} \left(A^{j_i}_{\alpha_{i-1},\alpha'_i}\right)^\dagger \cdot A^{j_i}_{\alpha_{i-1},\alpha_i} = \delta_{\alpha'_i,\alpha_i}\,, \tag{5.15}$$

$$\sum_{j_i}\sum_{\alpha_i} B^{j_i}_{\alpha'_{i-1},\alpha_i} \cdot \left(B^{j_i}_{\alpha_{i-1},\alpha_i}\right)^\dagger = \delta_{\alpha'_{i-1},\alpha_{i-1}}\,, \tag{5.16}$$

which is inherited from the orthogonality of the basis states in Eq. (5.2).

Up to this point, we have gained nothing by introducing MPSs. The key to their efficacy is the entropy area law, which was discussed in Sec. 3.3. We recall that the von Neumann entropy S of any mixed state within the ground state of a local, gapped, one-dimensional, Hamiltonian does not exceed a certain threshold S_{max}, which itself is independent of the system size. Since S is calculated within an MPS at any virtual bond i as

$$S = -\sum_{\alpha_i} \Lambda^2_{\alpha_i} \ln \Lambda^2_{\alpha_i}\,, \tag{5.17}$$

the number of states α_i one needs to keep to accurately represent a ground state is also bounded from above according to Eq. (3.30). The truncation process by which the dominant states are selected was described in Sec. 4.5. So just as for AKLT states, Eq. (3.5), the size of an MPS that stores a ground state grows linearly with system size, instead of exponentially, as it would if we used Eq. (5.1). Verstraete et al. describe this in more detail in Ref. [103].

For an in-depth examination of MPSs and their properties, see Refs. [104, 105].

5.2 Matrix product operators

Consider the MPS given by

$$|\psi\rangle = \sum_{\substack{j_1,...,i_N \\ \alpha_1,...,\alpha_{N-1}}} A^{j_1}_{\alpha_1} A^{j_2}_{\alpha_1,\alpha_2} A^{j_3}_{\alpha_2,\alpha_3} ... A^{j_N}_{\alpha_{N-1}} |j_1, ..., j_N\rangle \ , \tag{5.18}$$

where the Schmidt-coefficients in Eq. (5.12) were attached to the right tensor. We now look for a representation of many-body operators $\hat{O}$ in the form of matrix product operators (MPOs), whose topology matches that of Eq. (5.18). A suitable form is

$$\hat{O} = \sum_{\substack{j_1,...,i_N \\ j'_1,...,i'_N \\ \beta_1,...,\beta_{N-1}}} W^{j_1,j'_1}_{\beta_1} W^{j_2,j'_2}_{\beta_1,\beta_2} W^{j_3,j'_3}_{\beta_2,\beta_3} ... W^{j_N,j'_N}_{\beta_{N-1}} |j'_1, ..., j'_N\rangle \langle j_1, ..., j_N| \ , \tag{5.19}$$

which is illustrated graphically in Fig. 5.2. Crosswhite et al. [106, 107] have devised a

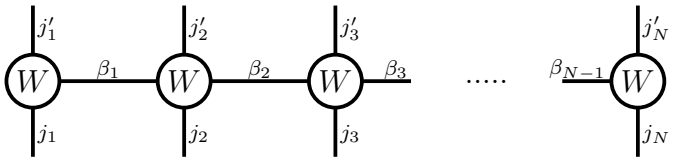


Figure 5.2: Matrix product operator

scheme to systematically convert a Hamiltonian, such as that of the anisotropic Heisenberg model,

$$H = \sum_{i=1}^{N-1} \left[\frac{J}{2} \left(S^+_i S^-_{i+1} + S^-_i S^+_{i+1} \right) + \Delta S^z_i S^z_{i+1} \right] - h \sum_{i=1}^{N} S^z_i \, , \tag{5.20}$$

into the form of Eq. (5.19) using finite-state machine (FSM). The first step is to interpret the set of all characters of H, meaning local field-operators with optional prefactors, as an alphabet Σ:

$$\Sigma = \left\{ \mathbb{1}, \frac{J}{2} S^+, \frac{J}{2} S^-, \Delta S^z, -h \, S^z, S^+, S^-, S^z \right\} . \tag{5.21}$$

Concatenation of multiple characters leads to a word w, such as

$$w = \mathbb{1}_1 \otimes \mathbb{1}_2 \otimes \Delta \, S^z_3 \otimes S^z_4 \otimes \mathbb{1}_5 \tag{5.22}$$

for longitudinal spin interaction between site 3 and 4 on a chain of 5 sites. The set of all possible words made out of Σ is called Σ^*. However, most of its elements are actually not physical, e.g.,

$$w = \mathbb{1}_1 \otimes S^+_2 \otimes S^+_3 \otimes S^+_4 \otimes \mathbb{1}_5 \, . \tag{5.23}$$

A regular language L is a subset of Σ^*, in our case, the set of all words which lead to a valid term in the Hamiltonian of Eq. (5.20).

The aforementioned FSM, which we define as A, is what generates all of the elements of L. It consists of a finite set of internal states Q, a transition function $\delta : Q \times \Sigma \to Q$, an initial state $q_0 \in Q$, and a subset of final or accepted states $F \subset Q$. Any possible word $w \in \Sigma^*$ is iterated over by considering each character separately and changing the FSM's state. If, after scanning the word completely, A is in an accepted state, w is an element of L. Fig. 5.3 depicts A as a directed graph for the Heisenberg model.

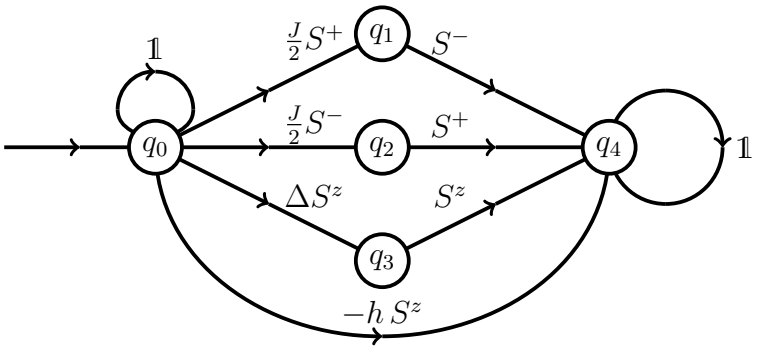


Figure 5.3: Finite-state machine for the anisotropic Heisenberg model in one dimension.

When a word is queried, one has to imagine A being initialized via the arrow on the left, which leads to the initial state q_0. At first, an arbitrary number of identities can be inserted, as shown by the curved arrow labeled with $\mathbb{1}$. Once one of the characters constituting a two-point correlation $(\frac{J}{2}S^+, \frac{J}{2}S^-, \Delta S^z)$ is inserted, a transition to an intermediate state (q_1, q_2, q_3) must take place. The next character must then be the counterpart (S^-, S^+, S^z), which leads to the final state q_4. The term that couples a spin locally to a magnetic field, $-h\,S^z$, leads to a direct transition from q_0 to q_4. Once in the final state, a word is only allowed to be appended with more identities, as is illustrated by the circular arrow on the right.

The bulk tensors W in Eq. (5.19) can now be identified as matrices that mediate the transition from one state to another,

$$W_{\beta_{i-1},\beta_i} = \begin{pmatrix} \mathbb{1} & \frac{J}{2}S^+ & \frac{J}{2}S^- & \Delta S^z & -h\,S^z \\ 0 & 0 & 0 & 0 & S^- \\ 0 & 0 & 0 & 0 & S^+ \\ 0 & 0 & 0 & 0 & S^z \\ 0 & 0 & 0 & 0 & \mathbb{1} \end{pmatrix}, \tag{5.24}$$

where the first and the last tensors are given by the first row and the last column of

Eq. (5.24), respectively:

$$W_{\beta_1} = \begin{pmatrix} \mathbb{1} & \frac{J}{2}S^+ & \frac{J}{2}S^- & \Delta S^z & -h\,S^z \end{pmatrix}, \tag{5.25}$$

$$W_{\beta_{N-1}} = \begin{pmatrix} -h\,S^z & S^- & S^+ & S^z & \mathbb{1} \end{pmatrix}^T. \tag{5.26}$$

One can adapt this scheme to any other local, one-dimensional Hamiltonian.

If states q_i are treated on equal footing with sets of quantum numbers, the W can also be stored in block-sparse format, as listed in Table 4.1. Note also the difference between virtual bonds α_i in MPSs and virtual bonds β_i in MPOs. The former encode entanglement between the subparts of a wavefunction, whereas the latter represent states in an FSM.

As a final remark, we describe how to convert simple operators $\hat{O}$, which consist of only one summand, into their MPO representation. One simply needs to add dummy indices β_i with a dimension of one, as we show for the term given in Eq. (5.22):

$$\hat{O} = \sum_{j_i, j_i', \beta_i} \mathbb{1}^{j_1,j_1'}_{\beta_1} \mathbb{1}^{j_2,j_2'}_{\beta_1,\beta_2} \left(\Delta\, S^+\right)^{j_3,j_3'}_{\beta_2,\beta_3} \left(S^-\right)^{j_4,j_4'}_{\beta_2,\beta_3} \mathbb{1}^{j_5,j_5'}_{\beta_4} \left|j_1', j_2', j_3', j_4', j_5'\right\rangle \left\langle j_1, j_2, j_3, j_4, j_5\right| . \tag{5.27}$$

5.3 Expectation values

We now turn to one of the most elementary operations on a wave function $|\psi\rangle$, which is the calculation of an expectation value $\langle\hat{O}\rangle = \langle\psi|\,\hat{O}\,|\psi\rangle$. The operator $\hat{O}$ can be an observable we want to measure, such as the Hamiltonian or a local density, or just the identity, in which case the norm is evaluated. The overall functional can be represented graphically as a tensor network, illustrated in Fig. 5.4. The first, second, and third rows of

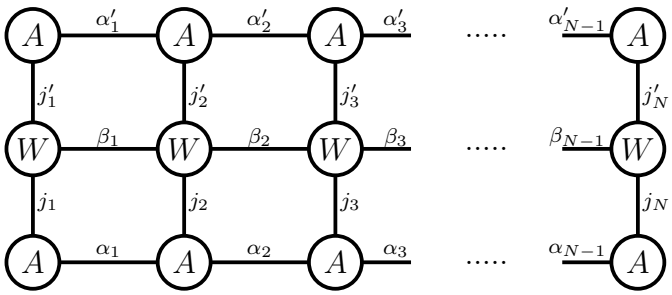
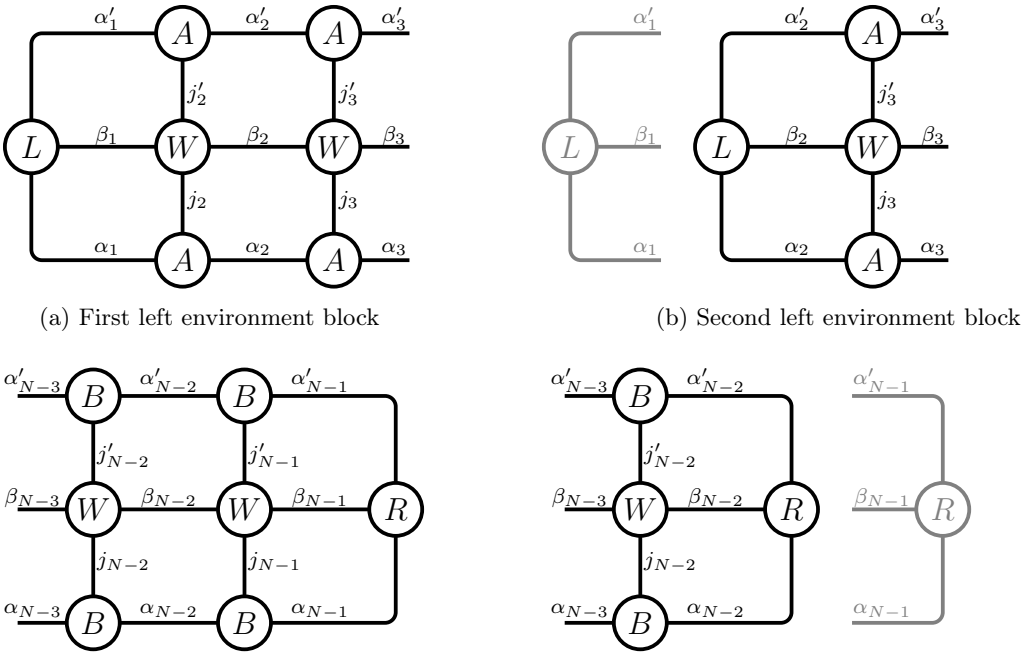


Figure 5.4: $\langle\psi|\,\hat{O}\,|\psi\rangle$ as a tensor network.

tensors represent the MPS $|\psi\rangle$, the MPO $\hat{O}$, and the adjoint MPS $\langle\psi|$, respectively. The contraction of the network takes place by successive calculation of rank-3 environment tensors from either the left or from the right, as shown in the first steps in Fig. 5.5.

In the following, we consider the contraction from the left in more detail: The first left



(a) First left environment block

(b) Second left environment block

(c) First right environment block

(d) Second right environment block

Figure 5.5: Calculation of environment blocks

environment block L is defined as

$$L_{\alpha'_1,\beta_1,\alpha_1} = \sum_{j'_1,j_1} \left[\left(A^{j'_1}_{\alpha'_1} W^{j'_1,j_1}_{\beta_1} \right) A^{j_1}_{\alpha_1} \right] , \tag{5.28}$$

whereas all subsequent blocks are calculated using the preceeding L,

$$L_{\alpha'_i,\beta_i,\alpha_i} = \sum_{j'_i,j_i,\alpha'_{i-1},\beta_{i-1},\alpha_{i-1}} \left\{ \left[\left(L_{\alpha'_{i-1},\beta_{i-1},\alpha_{i-1}} A^{j'_i}_{\alpha'_{i-1},\alpha'_i} \right) W^{j'_i,j_i}_{\beta_{i-1},\beta i} \right] A^{j_i}_{\alpha_{i-1},\alpha_i} \right\} , \tag{5.29}$$

with all of them operating on left-normalized MPS tensors A, meaning that the Schmidt coefficients are attached to the right. The individual steps of Eq. (5.29) are depicted in Fig. 5.6. If the number of states in an MPS is m, the number of states in the MPO is n, and the size of the local Hilbert spaces is d, one can easily see that the three contractions scale as $\mathcal{O}\left(m^3\, n\, d\right)$, $\mathcal{O}\left(m^2\, n^2\, d^2\right)$, and $\mathcal{O}\left(m^3\, n\, d\right)$, respectively.

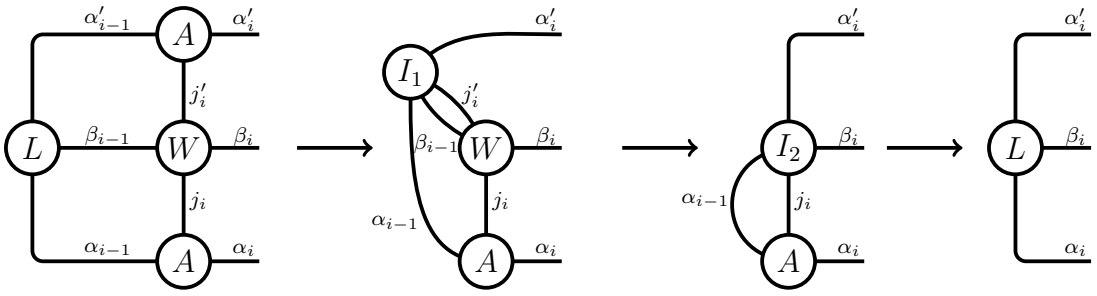


Figure 5.6: Step by step contraction of a left environment block with the intermediate tensors I_1 and I_2.

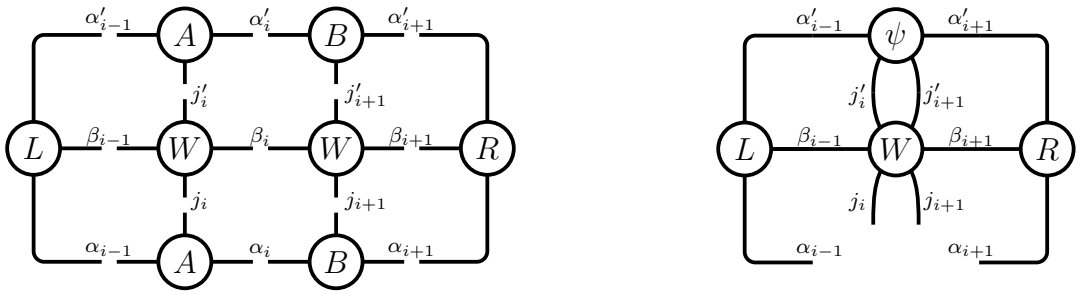
The right environment blocks R are calculated accordingly from the other end of the tensor network, but with right-normalized MPS-tensors B. While the normalization makes no difference for the calculation of expectation values, it is useful to set up the definitions of L and R this way for considerations in Sec. 5.4. In both cases, the contractions clearly continue until the other end of the tensor network is reached, and the final result is a single number, $\langle\psi|\,\hat{O}\,|\psi\rangle$. Furthermore, for an efficient implementation, the blocks gathered along the way should be cached, so they can be reloaded for other expectation values which have a subset of the tensor network in common.

5.4 Variational optimization

Given the representations of wave functions as MPSs and Hamiltonians as MPOs, variational calculations in the context of tensor networks can be carried out. The overall goal is to find the ground state, which is the normalized $|\psi\rangle$ that minimizes the energy $E = \langle\psi|\,H\,|\psi\rangle$ with respect to a Hamiltonian H. If $|\psi\rangle$ is initialized with a starting value, for instance, a product state, the ground state of H could be determined by optimizing $|\psi\rangle$ as a whole, as was shown in Sec. 2.1. Since the numerical effort increases exponentially with system size, this approach is unsuitable for treating systems that scale to the thermodynamic limit. Instead, we take advantage of the newly introduced tensor structure and again consider the full network of Fig. 5.4 for $\hat{O} = H$ and a mixed canonical MPS of the form

$$|\psi\rangle = \sum_{\substack{j_1,\ldots,i_N \\ \alpha_1,\ldots,\alpha_{N-1}}} A^{j_1}_{\alpha_1} \ldots A^{j_{i-1}}_{\alpha_{i-2},\alpha_{i-1}} A^{j_i}_{\alpha_{i-1},\alpha_i} B^{j_{i+1}}_{\alpha_i,\alpha_{i+1}} B^{j_{i+2}}_{\alpha_{i+1},\alpha_{i+2}} \ldots B^{j_N}_{\alpha_{N-1}} |j_1, \ldots, j_N\rangle \ . \tag{5.30}$$

Fig. 5.4 is now contracted both from the left and from the right, as shown in Fig. 5.5, until the two sites i and $i+1$ are isolated, which leads to the structure depicted in Fig. 5.7(a).



(a) Tensor network contraction excluding two sites.

(b) Core operation $H_{\text{eff}} \cdot \psi$.

Figure 5.7: Variational optimization of matrix product states.

Instead of optimizing the whole wave function, we have thus fixed most of its degrees of freedom and have applied it to two sites. The omitted tensors A and B together form a two-site tensor ψ, which is to be seen as the initial vector for an iterative diagonalization.

Because of the mixed canonical form, the norm of the two-site tensor is equal to the norm of the overall wave function. If we combine the remaining two tensors of the MPO as $W_{\beta_{i-1},\beta_{i+1}} = W_{\beta_{i-1},\beta_i} W_{\beta_i,\beta_{i+1}}$, the effective Hamiltonian

$$H_{\text{eff}} = L_{\alpha'_{i-1},\beta_{i-1},\alpha_{i-1}} \cdot W^{j_i,j'_i,j_{i+1},j'_{i+1}}_{\beta_{i-1},\beta_{i+1}} \cdot R_{\alpha'_{i+1},\beta_{i+1},\alpha_{i+1}} \tag{5.31}$$

covers the subsector of the Hilbert space in which we want to find the ground state. The optimal ψ can now be found using an iterative diagonalization algorithm, such as the Davidson algorithm discussed in Sec. 2.1. Its core operation which is the algorithm's overall major bottleneck is $H_{\text{eff}} \cdot \psi$ and is depicted as a network in Fig. 5.7(b). Just like in Section 5.3, we consider the scaling behavior of the individual contractions, but this time with respect to the sequence of operators $\psi \to L \to H \to R$, yielding $\mathcal{O}\left(m^3\, n\, d^2\right)$, $\mathcal{O}\left(m^2\, n^2\, d^4\right)$, and $\mathcal{O}\left(m^3\, n\, d^2\right)$, respectively. Thus the leading scaling of the optimization is cubic in the number of states. The actual processing takes place by the graph-guided contraction, as was discussed in Section 4.6. Note that, due to its size, H_{eff} itself is never calculated. Furthermore, a single-site optimization has significant deficiencies: While optimizing one site rearranges weights within the current basis of a wave function, the two-site algorithm actually explores new subsectors of the Hilbert space through transitions between the physical bonds j_i and j_{i+1}.

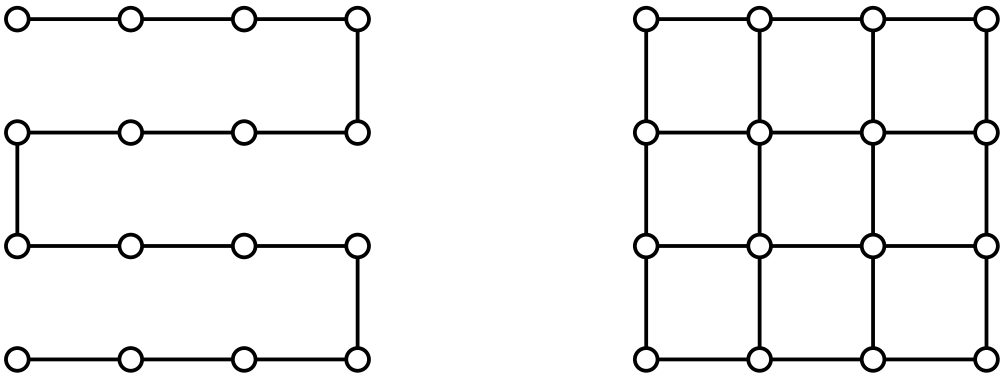
Once the optimization is finished, the lost virtual bond α_i is retrieved through the truncated decomposition described in Section 4.5, which converts ψ back into A and B, but with more optimal states along the α_i-bond. The dimension of α_i is determined by the aforementioned maximum number of states. The Schmidt coefficients are then attached to the left or the right tensor, depending on which two sites the next optimization step is to act. If, for instance, we want to make a step to the right, we attach Σ_{α_i} to the right MPS-tensor, calculate the new environment block $L_{\alpha'_i,\beta_i,\alpha_i}$, reload $R_{\alpha'_{i+2},\beta_{i+2},\alpha_{i+2}}$, and perform the iterative diagonalization for $i+1$ and $i+2$. When the right boundary is reached, we reverse direction and move all the way to the left and repeat the process. This sweeping back and forth, which scales only linearly with system size instead of exponentially for bare exact diagonalization, is done until the overall energy no longer decreases, and the resulting MPS is the best approximation for the ground state within the given maximum bond dimension. If the convergence analysis is done properly, one can then, in addition, increase the bond dimension from sweep to sweep. Then the *truncation error*, which is the weight of the states that are dropped during truncation, will decrease steadily until it becomes zero, the area law kicks in, all of the entropy is captured by the virtual states, and the MPS represents the physical ground state of the Hamiltonian.

This process of variational optimization of MPSs is a reformulation of the DMRG invented by S. R. White [37, 38]. Due to their equivalence, both terms can be used synonymously.

6 Two-dimensional systems

6.1 Projected entangled pair states

The MPSs introduced in Chapter 5.1 can not only be used to approximate wave functions on chains, but also on lattices, as shown in Fig. 6.1(a). While it is, in principle, possible to



(a) Matrix product state (b) Projected entangled pair state

Figure 6.1: Tensor network states on a 4x4 square lattice.

map two dimension onto one, this involves a complication that is best explained through the scaling of the entropy: Consider the best-case scenario: that the ground state of the Hamiltonian exhibits an area law. This means that the block entropy of any subsystem is proportional to the size of its boundary. The boundary, however, is defined by the set of sites on which an operator acts that connects the subsystem to the environment. If we reexamine Fig. 6.1(a), we see that vertical, short-ranged correlations, such as $S_i^+ S_{i+1}^-$ in spin systems, need to be mapped onto long-range correlations in one dimension. Increasing width thus leads to a linear increase of the range, which leads to a linear increase of the size of the effective boundary and, consequentially, the entropy. The number of states along virtual bonds necessary to represent the ground state reliably then grows exponentially due to Eq. (3.30). Although coupled chains and more general two-dimensional systems of small widths have been studied extensively using the DMRG [64, 108–122], the thermodynamic limit will probably remain inaccessible due to this linear increase of entropy and the corresponding exponential increase of computational effort.

A solution was presented by Verstraete et al. in the form of *projected entangled pair states* (PEPS), which are the natural generalization of MPSs to two dimensions [51]. An example is illustrated in Fig. 6.1(b), which is nothing but the aerial view of the cyclic tensor network in Fig. 4.2(b). Since the system is not multipartite, meaning that cutting one bond does not divide it into two, the Schmidt decomposition of Eq. (5.2) cannot be applied to transform a generic quantum state into its tensor network representation.

Instead, each bond between adjacent sites is associated with a maximally entangled state

$$|\phi\rangle = \sum_{k=1}^{D} |k,k\rangle \ , \tag{6.1}$$

in a virtual Hilbert space, where D is the maximum bond dimension. Conversely, this means that a bulk site i is connected to four virtual bonds u_i, d_i, l_i, and r_i. If we define the projector Q_i, which maps virtual bonds at site i to the physical bond j_i, we can define the following rank-5 PEPS-tensor

$$A_i = A^{j_i}_{u_i,d_i,l_i,r_i} = \langle j_i | \, Q_i \, | u_i, d_i, l_i, r_i \rangle \ , \tag{6.2}$$

whose MPS-equivalent was defined in Eq. (5.10). Tensors at the edges or the corners of the lattice are rank-4 and rank-3, accordingly. There is no straightforward way of writing a PEPS as an explicit equation due to the two-dimensional arrangement of tensors, which is why we define it simply as

$$|\psi\rangle = \sum_{j_1,...,j_N} \mathcal{F}\left(A_1 \, A_2 \, ... \, A_N\right) |j_1, ..., j_N\rangle \, , \qquad \mathcal{F}\left(A_i \, A_2 \, ... \, A_N\right) = \psi_{j_1,...,j_N} \, , \tag{6.3}$$

for a lattice of N sites, where $\mathcal{F}$ is a function that contracts common virtual indices in PEPSs according to the lattice structure. Note that the construction of A_i via Q_i is mainly academic; in practical simulations, a PEPS will be initialized as a product state with virtual indices of dimension one. As the optimization proceeds, the dimensions then grow and lead to states which are, in general, not maximally entangled.

Because of the cyclic structure of a PEPS, multiple paths lead from one physical site to another, which means that a single virtual bond cannot be clearly associated with an entanglement. Instead, one needs to choose a cluster of physical Hilbert spaces and bundle all virtual indices which connect them to the environment. Fig. 6.2 illustrates the simplest example, which is the calculation of the entropy of a single site:

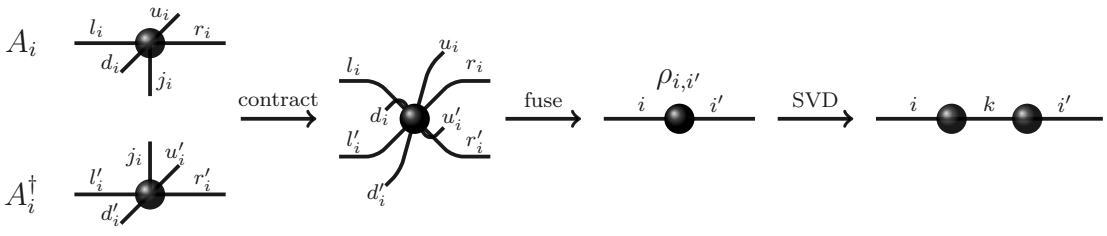


Figure 6.2: Entanglement of a PEPS tensor with its environment, encoded in the index k.

Eq. (6.2) is contracted with its adjoint along the physical index, which leads to a rank-8 reduced tensor. Afterwards, both the primed and unprimed indices are fused (Chapter 4.4), which yields the reduced density matrix $\rho_{i,i'}$ of site i. In the end, we determine

the eigenvalue decomposition of $\rho_{i,i'}$, which can be carried out by an SVD (Chapter 4.5). The resulting eigenvalues, parameterized by k, can be used to calculate the von Neumann entropy S between the considered site and the rest of the PEPS. Note how, due to the maximum bond dimension D, which was fixed in Eq. (6.1), PEPSs exhibit an area law by construction; the value of S for a single PEPS tensor or a larger cluster does not change if we add new sites to the system.

PEPSs are promising candidates for describing ground states in two-dimensional quantum systems [54, 56]. Their introduction, however, leads to two complications, which will be discussed later on. First, the exact calculation of expectation values $\langle\psi| O |\psi\rangle$ is NP-hard, which is why it has to be done approximately, and, second, due to the high rank of bulk-tensors, PEPS-based algorithms scale with a high power of the number of states.

6.2 Projected entangled pair operators

Analogously to the construction of MPOs which act on MPSs, we now look for a representation of local operators which is similar to that of a PEPS. Consider an electronic many-body system on a two-dimensional lattice that is described by the Hubbard Hamiltonian

$$\mathcal{H} = -t \sum_{\langle i,j\rangle,\sigma} \left(c^\dagger_{i,\sigma}\, c_{j,\sigma} + \text{h.c.} \right) + U \sum_i c^\dagger_{i,\uparrow} c^\dagger_{i,\downarrow} c_{i,\downarrow} c_{i,\uparrow} \,,$$

with fermionic creation and annihilation operators

$$\begin{aligned} c^\dagger_\uparrow &= |\uparrow\rangle \langle 0| + |\uparrow\downarrow\rangle \langle\downarrow| \ , \\ c_\uparrow &= |0\rangle \langle\uparrow| + |\downarrow\rangle \langle\uparrow\downarrow| \ , \\ c^\dagger_\downarrow &= |\downarrow\rangle \langle 0| - |\uparrow\downarrow\rangle \langle\uparrow| \ , \\ c_\downarrow &= |0\rangle \langle\downarrow| - |\uparrow\rangle \langle\uparrow\downarrow| \ , \end{aligned}$$

acting on local Hilbert spaces spanned by the four basis states $\{|0\rangle, |\uparrow\rangle, |\downarrow\rangle, |\uparrow\downarrow\rangle\}$. Here $\langle i,j\rangle$ denotes nearest neighbors, σ the spin configuration, t the tunneling amplitude, and U the onsite repulsion. We wish to store this expression as a projected entangled pair operator (PEPO), which has the same topology as the PEPS to which it is supposed to be applied. To our knowledge, only Froewis et al. [123] have suggested an algorithm for the construction of PEPOs; in particular, they did this for long-ranged, bosonic Hamiltonians of the form

$$\mathcal{H} = \sum_{i<j} c_{i,j} X_i \otimes Y_i \,.$$

Here we consider local fermionic systems, which would lead to N-point correlations due to an inverse Jordan-Wigner transformation if their scheme were to be generalized to fermions directly. Therefore, we will choose a different approach.

We start by expanding the full Hamiltonian for a lattice of a given width and height. For example, for a 4×4 lattice with open boundary conditions, the full operator reads, schematically,

$$\begin{aligned}
\mathcal{H} = &\left(U c_\uparrow^\dagger c_\downarrow^\dagger c_\downarrow c_\uparrow\right)_1 \otimes \mathbb{1}_2 \otimes \mathbb{1}_3 \otimes \ ... \ \otimes \mathbb{1}_{16} \\
&+ \mathbb{1}_1 \otimes \left(U c_\uparrow^\dagger c_\downarrow^\dagger c_\downarrow c_\uparrow\right)_2 \otimes \mathbb{1}_3 \otimes \ ... \ \otimes \mathbb{1}_{16} \\
&... \\
&+ \mathbb{1}_1 \otimes \mathbb{1}_2 \otimes \ ... \ \otimes \mathbb{1}_{15} \otimes \left(U c_\uparrow^\dagger c_\downarrow^\dagger c_\downarrow c_\uparrow\right)_{16} \\
&+ \left(-t\mathcal{P} c_\uparrow^\dagger\right)_1 \otimes (c_\uparrow)_2 \otimes \mathbb{1}_3 \otimes \ ... \ \otimes \mathbb{1}_{16} \\
&+ \mathbb{1}_1 \otimes \left(-t\mathcal{P} c_\uparrow^\dagger\right)_2 \otimes (c_\uparrow)_3 \otimes \mathbb{1}_4 \otimes \ ... \ \otimes \mathbb{1}_{16} \\
&... \\
&+ \left(-t\mathcal{P} c_\uparrow^\dagger\right)_1 \otimes \mathcal{P}_2 \otimes \mathcal{P}_3 \otimes \mathcal{P}_4 \otimes (c_\uparrow)_5 \otimes \mathbb{1}_6 \otimes \ ... \ \otimes \mathbb{1}_{16} \\
&...
\end{aligned}$$

where the parity operator

$$\mathcal{P} = |0\rangle\langle 0| - |\uparrow\rangle\langle\uparrow| - |\downarrow\rangle\langle\downarrow| + |\uparrow\downarrow\rangle\langle\uparrow\downarrow|$$

encodes the effect of the fermionic sign. (Alternatively, one could incorporate the sign into the PEPS through fermionic swap gates [49, 60].) The subscripts denote the physical sites on which the operators act, whereas the enumeration is chosen according to the scheme depicted in Fig. 6.3.

Inspired by the construction of MPOs via FSMs, which was discussed in Sec. 5.2, we now conceptualize $\mathcal{H}$ as a regular expression. In particular, for each unique sequence of operators on one site, we substitute one symbol ($U c_\uparrow^\dagger c_\downarrow^\dagger c_\downarrow c_\uparrow \to$ U, $-t\mathcal{P} c_\uparrow^\dagger \to$ A, $c_\uparrow \to$ B, $-t\mathcal{P} c_\downarrow^\dagger \to$ C, $c_\downarrow \to$ D, $\mathbb{1} \to$ I, $\mathcal{P} \to$ P). Tensor products connecting different local Hilbert spaces are then interpreted as concatenations, whereas the sums become unions. In this way, all symbols form an alphabet, each summand of the many-body Hamiltonian becomes a word, and the set of all words forms a regular language:

$$H = \{ \text{UIIIIIIIIIIIIII, IUIIIIIIIIIIIII, ... , IIIIIIIIIIIIIU, ABIIIIIIIIIIIII,}$$
$$\text{IABIIIIIIIIIIII, ... , APPPBIIIIIIIIII, ...} \} \quad .$$

Here we drop the subscripts - they are determined by the position of a symbol in the sequence.

The PEPO we want to form will be a set of interconnected FSMs, which generate all of the words of the language in a two-dimensional fashion. In order to do this, we connect adjacent sites through directed signalling channels, as depicted in Fig. 6.3. The flow of information is defined as going upwards and to the right. Information at the upper boundary is dropped, while information at the right boundary is redirected to the top, which makes the rightmost vertical channels of the lattice the trunk. The upper-right corner (site 16 in Fig. 6.3) is thus the sink, where all the information passed by the FSMs is gathered, and it is determined whether a string of symbols is actually a part of the language and, hence, is a meaningful term of the Hamiltonian. For example, channel 5 (21) reports what symbols have occured on sites 5 and 6 (3, 7 and 11). Channel 23, as a trunk channel, carries information from the entire block below, i.e., from all sites from 1 to 8. In this way, each FSM can be seen as an associative tensor or table, in which a set of incoming and outgoing states forms a key, the associated symbol represents their value, and the pair of both is an element. The rank is equal to the coordination number, which is two at the corners, three at the edges, and four otherwise. As an example, the FSM on site 6 is schematically described in Table 6.1.

We define three distinguished states, s_i, s_f and s_P, which are the initial state, the final state and the parity state. Here s_i indicates that, so far, only identities ("I") have appeared, and the first occurence of a non-trivial symbol is pending. Now s_f states the complement, namely that a valid combination of non-trivial symbols has already appeared and, from now on, only identities are allowed to be attached to the word. Finally, s_P takes the parity operators from the Jordan-Wigner transformation into account and connects

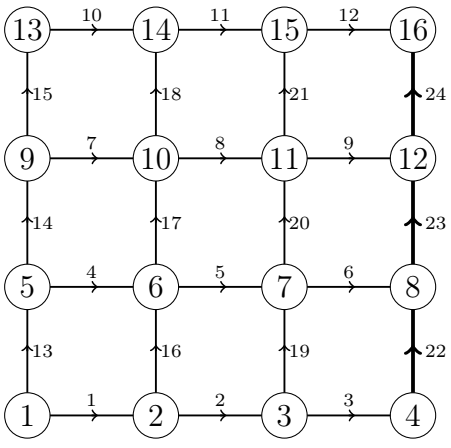


Figure 6.3: Sites and signalling channels in between for a 4×4 lattice

vertical hopping terms in a way that will be elucidated via an example later on. These states, together with the two trivial symbols "I" and "P", are used to initialize the PEPO as depicted in Tables 6.2 and 6.3. Note how there are no elements with two incoming final states s_f, which prevents different words from mixing with each other.

After all of these preparations, the Hamiltonian can now be converted into a PEPO by recasting every word into a set of non-trivial pairs, with the first element being a non-trivial symbol and the second the index of the site it acts on. Each pair is then inserted into the PEPO by adding an element to the tensor of the respective site. In this way, the entire PEPO can be systematically constructed word for word.

For example, the word "IIIIIUIIIIIIIII" translates as $\{(U, 6)\}$ and is incorporated into the PEPO by adding the element $\{(s_i, s_i, s_f, s_f), U\}$ to the tensor at site 6 in accordance

Table 6.1: Finite state machine for site 6 implemented as an associative tensor. For the incoming states $s_{4,i}$ from channel 4 and $s_{16,i}$ from channel 16, the symbol σ_i is inserted into the current word and $s_{5,i}$ and $s_{17,i}$ are emitted via channels 5 and 17.

C 4	C 16	C 5	C 17	S 6
$s_{4,1}$	$s_{16,1}$	$s_{5,1}$	$s_{17,1}$	σ_1
$s_{4,2}$	$s_{16,2}$	$s_{5,2}$	$s_{17,2}$	σ_2
$s_{4,3}$	$s_{16,3}$	$s_{5,3}$	$s_{17,3}$	σ_3
...				

Table 6.2: Initial elements of a bulk tensor/FSM. Here C I_l and C I_b are the two incoming channels from left and bottom and C O_r and C O_t are the outgoing channels to the right and top.

C I_l	C I_b	C O_r	C O_t	S
s_i	s_i	s_i	s_i	I
s_i	s_f	s_i	s_f	I
s_f	s_i	s_f	s_i	I
s_i	s_P	s_i	s_P	I
s_f	s_P	s_f	s_P	I
s_P	s_i	s_P	s_P	P

Table 6.3: Initial elements of a trunk tensor/FSM. Here C I_l and C I_b are the two incoming channels from left and bottom, O_t is the outgoing channel to the top.

C I_l	C I_b	C O_t	S
s_i	s_i	s_i	I
s_f	s_i	s_f	I
s_i	s_f	s_f	I
s_f	s_P	s_f	I
s_P	s_i	s_P	P

with Table 6.1. The remaining 15 identities not mentioned in the set are taken care of by the previous initialization.

The next type of words are those which represent horizontal hopping terms, such as "IIIIIABIIIIIIIII", which is shown in Fig. 6.4. The set of pairs for this case reads $\{(A,6),(B,7)\}$ and the elements which need to be attached to the tensors of site 6 and 7 are $\{(s_i,s_i,\{(A,6)\},s_f),A\}$ and $\{(\{(A,6)\},s_i,s_f,s_f),B\}$ respectively. This means that

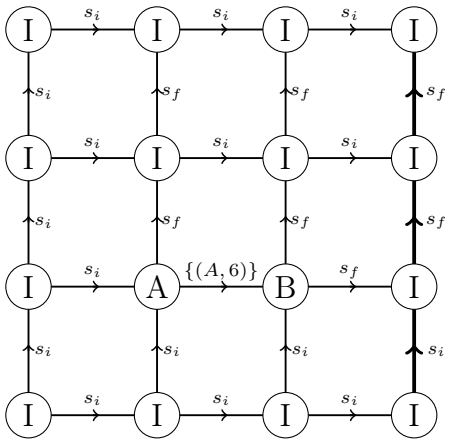


Figure 6.4: Graphical depiction of the word "IIIIIABIIIIIIIII" embedded into the PEPO.

at site 6, a word is initialized by inserting the symbol "A". A final state is emitted to the top, meaning that no other non-trivial symbol is expected above, while the intermediate state $\{(A,6)\}$ is sent to the right. At site 7, the aforementioned state is received from the left, the second symbol B is attached to the word and a final state is emitted to both the top and the right.

Finally, we consider terms that span multiple rows and, as a concrete example, "IAPPP-BIIIIIIIII", as shown in Fig. 6.5, which describes vertical hopping between site 2 and 6. Since the parity operators are already taken into account through proper initialization,

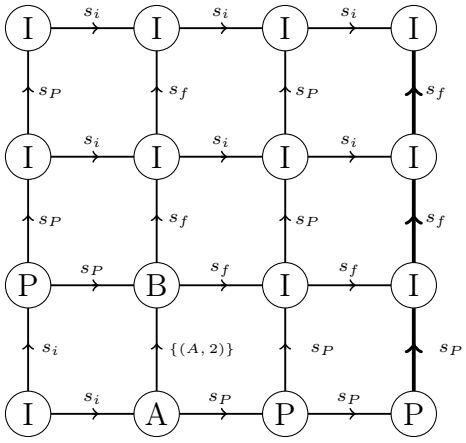


Figure 6.5: Graphical depiction of the word "IAPPPBIIIIIIIII" embedded into the PEPO

the term is effectively local and can be included by just two pairs $\{(A,2),(B,6)\}$ and the

according tensor elements $\{(s_i, s_P, \{(A,2)\}), A\}$ and $\{(s_P, \{(A,2)\}, s_f, s_f), B\}$ at sites 2 and 6, respectively.

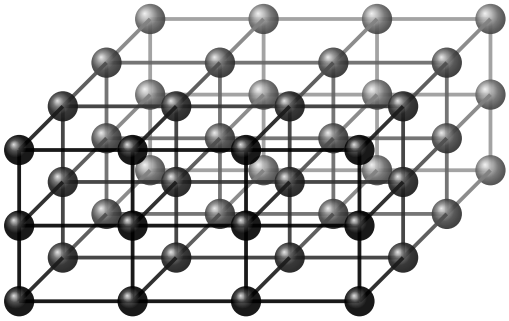
Adding a new element to a tensor only takes place if it does not yet exist. In the case of non-local Hamiltonians, which are not covered in this thesis, elements can already be present due to a previous update from another term, in which case, one element can be used for two or more words. Furthermore, it is crucial to define intermediate states as unordered sets, meaning that $\{(A,i),(B,j)\} = \{(B,j),(A,i)\}$.

The three examples considered above cover all possible kinds of words of local Hamiltonians as well as how to insert them into the respective tensors. After the PEPO is fully assigned, all of the intermediate states within keys, which are symbolic, can be substituted by unique numbers so that the PEPO can be stored efficiently. (For the case of the Hubbard model, the number of states is exactly seven, which represent s_i, s_f, s_p, and the four non-trivial operators, respectively.) This numbering can then be translated into the corresponding quantum numbers, meaning the spin S, its z-component S_z, and the deviation from half filling C_z. To take full advantage of the SU(2)-spin symmetry, the PEPO-tensors then have to be compactified through Eq. (4.20). Finally, the symbols within the tensors are substituted by the original quantum-mechanical operators. The PEPO is now ready to be used for the tensor network algorithm of one's choice, such as variational optimization or imaginary time evolution.

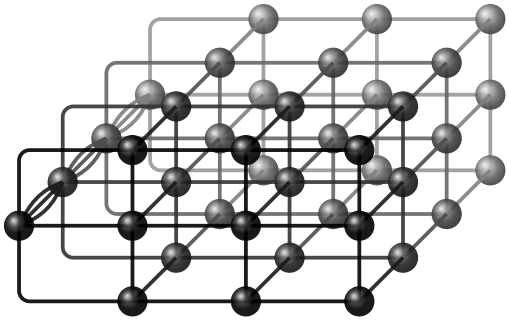
6.3 Expectation values

Given the representations of PEPSs and PEPOs, we are now interested in the calculation of expectation values, i.e., $\langle\psi|\hat{O}|\psi\rangle$, which is illustrated in Fig. 6.6 for a 4x4 lattice. Fig. 6.6(a) depicts the initial tensor network and can be seen as the generalization of Fig. 5.4 to two dimensions. The upper, middle, and lower planes are the PEPS, PEPO, and adjoint PEPS respectively, while the labels of indices have been dropped for the sake of clarity. The subsequent contraction steps in Figs. 6.6(b) to 6.6(e) are the two-dimensional equivalent of Figs. 5.5(a) and 5.5(b), in which one successively builds environments until the border is reached and the remaining tensors are contracted into a single number, yielding $\langle\psi|\hat{O}|\psi\rangle$ in Fig. 6.6(f).

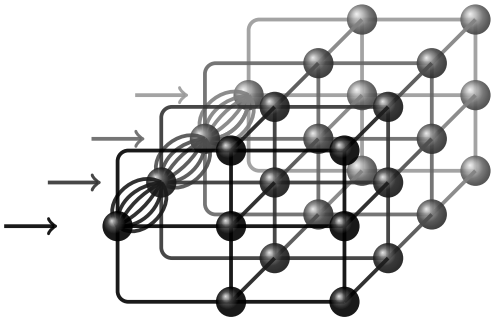
The difficulty with this straightforward and naive approach is obvious: In every new contraction step, the rank of intermediate tensors increases; this cannot be circumvented by any other conceivable order of contractions. The exact contraction of PEPS-based tensor networks is thus NP-hard with respect to system size and not feasible in real applications. Instead, we need to find an approximation for each environment, as shown in Fig. 6.7. Suppose we are given the arrangement of tensors on the left, which consist of environment blocks $E_{i-1,j}$ from a previous calculation, PEPS-tensors $A_{i,j}$, their adjoints $A^\dagger_{i,j}$, and PEPO-tensors $W_{i,j}$. The goal is to find the tensors $E_{i,j}$ on the right which, as a



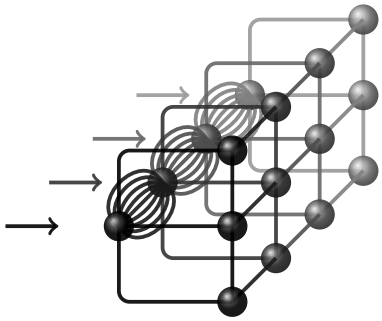
(a) $\langle\psi|\,\hat{O}\,|\psi\rangle$ as a tensor network.



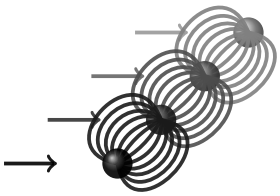
(b) Contraction of first environment.



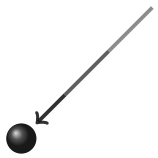
(c) Contraction of second environment.



(d) Contraction of third environment.



(e) Contraction of fourth environment.



(f) Contraction of remaining tensors.

Figure 6.6: Naive contraction of a PEPS-based tensor network.

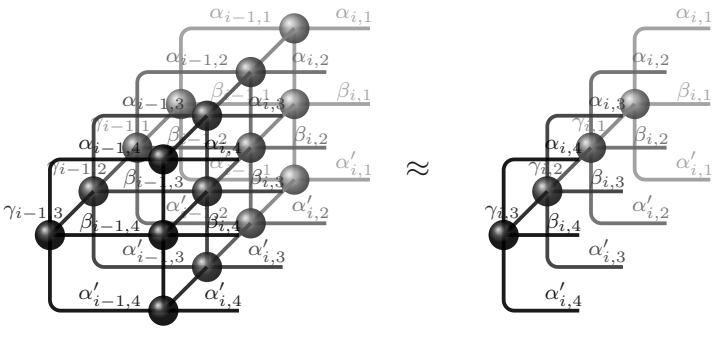


Figure 6.7: Approximation of environment.

whole, approximate the cluster on the left as well as possible [51] using cumulative indices $\gamma_{i,j}$ with a predetermined maximum bond dimension. If we interpret the former network

as a full vector

$$|\psi\rangle = \left(E_{i-1,1} \cdot A_{i,1} \cdot W_{i,1} \cdot A_{i,1}^{\dagger}\right) \cdot \ldots \cdot \left(E_{i-1,N} \cdot A_{i,N} \cdot W_{i,N} \cdot A_{i,N}^{\dagger}\right) , \tag{6.4}$$

and the latter as a truncated vector

$$|\tilde{\psi}\rangle = E_{i,1} \cdot E_{i,2} \cdot \ldots \cdot E_{i,N} , \tag{6.5}$$

the problem can be stated as finding the maximum of the fidelity

$$F = \frac{\langle\psi|\tilde{\psi}\rangle\langle\tilde{\psi}|\psi\rangle}{\langle\tilde{\psi}|\tilde{\psi}\rangle\langle\psi|\psi\rangle} , \tag{6.6}$$

which is, in general, done by setting

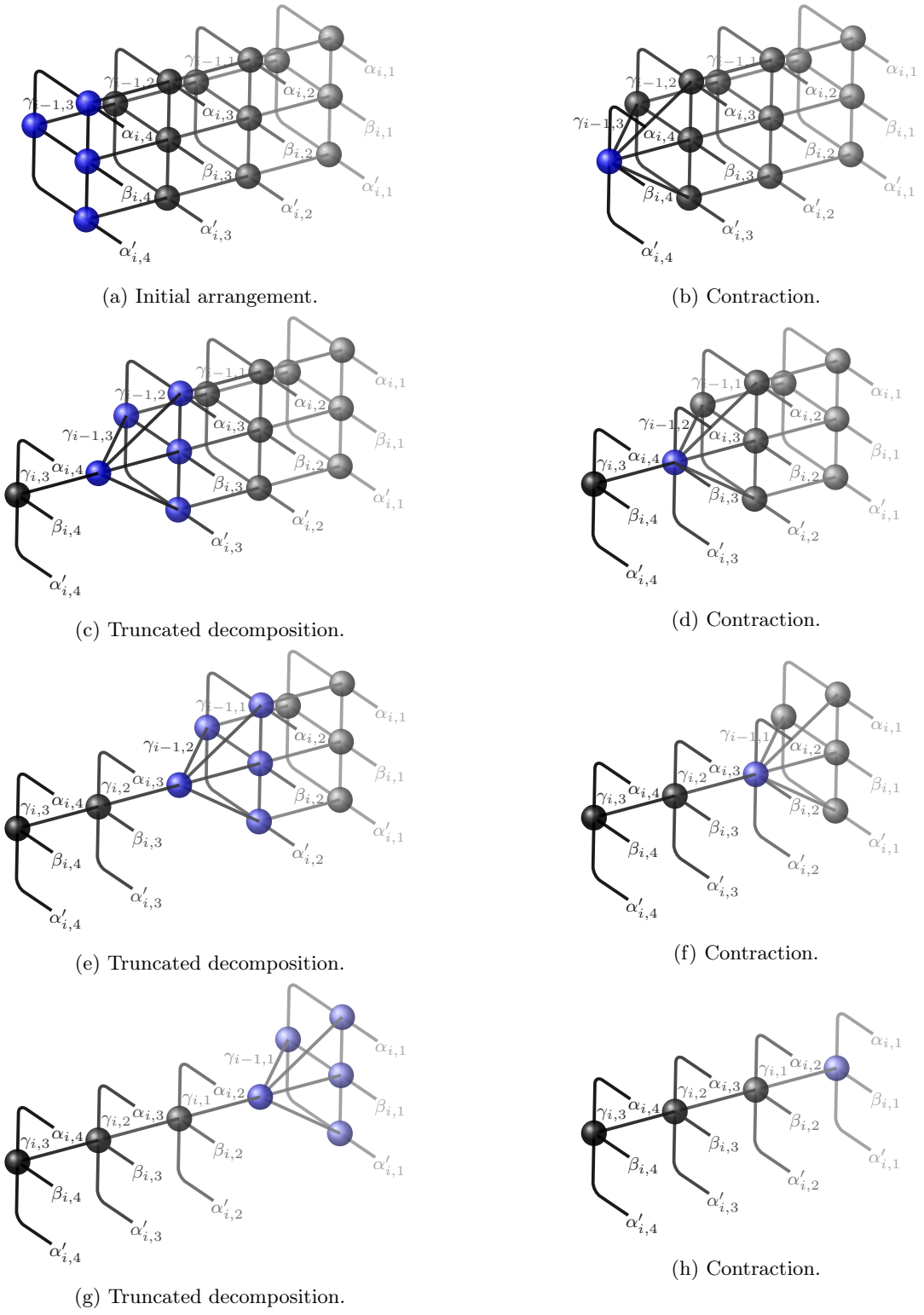
$$\langle\tilde{\psi}|\tilde{\psi}\rangle = \langle\tilde{\psi}|\psi\rangle. \tag{6.7}$$

Using Eq. (6.7) to find the optimal $|\tilde{\psi}\rangle$ as a whole is not possible, because it is fragmented into multiple tensors, and contracting all of them is what we want to avoid in the first place. Instead, we start with a trial vector, whose construction is depicted in Fig. 6.8. The cluster in Fig. 6.8(a) is the original $|\psi\rangle$, while the subsequent figures illustrate how to systematically forge a truncated environment with a bond dimension of one through alternating contractions and truncated SVDs. The outcome in Fig. 6.8(h) is then the starting point for an iterative algorithm. If the resulting trial vector has excluded certain modes within the full environment that cannot be reached because their overlap is zero, one can return to the initialization process and remove contributions orthogonal to the current approximation from it.

The optimization of $|\tilde{\psi}\rangle$ proceeds similarly to the DMRG: Most of its tensors are fixed by calculating both $\langle\tilde{\psi}|\tilde{\psi}\rangle$ and $\langle\tilde{\psi}|\psi\rangle$ partially from both sides. The individual steps are illustrated in Fig. 6.9, where we have switched to an aerial view, and the tensors F and G represent intermediate contractions for $\langle\tilde{\psi}|\tilde{\psi}\rangle$ and $\langle\tilde{\psi}|\psi\rangle$, respectively. Only two sites, say j and $j-1$, are set loose, which make it possible to determine the optimal, joint environment block, $X_{j,j-2} = E_{j,j-1} \cdot E_{j-1,j-2}$, by enforcing

$$\langle\tilde{\psi}|\tilde{\psi}\rangle / X_{j,j-2}^{\dagger} = \langle\tilde{\psi}|\psi\rangle / X_{j,j-2}^{\dagger} \tag{6.8}$$

in this reduced vector space. The calculation necessary to obtain the right-hand side is shown in Fig. 6.10, yielding the inhomogeneity B. The contraction is best performed by calculating the left and right halves of the cluster separately using the graph-guided contraction, followed by a pairwise multiplication of both results. Fig. 6.11 then illustrates the remaining steps necessary to set up the system of equations that leads to $X_{j,j-2}$: first, $F_{i,j}$ and $F_{i,j-2}$ are contracted, yielding the rank-4 tensor M. Subsequently, the cumulative



(a) Initial arrangement.

(b) Contraction.

(c) Truncated decomposition.

(d) Contraction.

(e) Truncated decomposition.

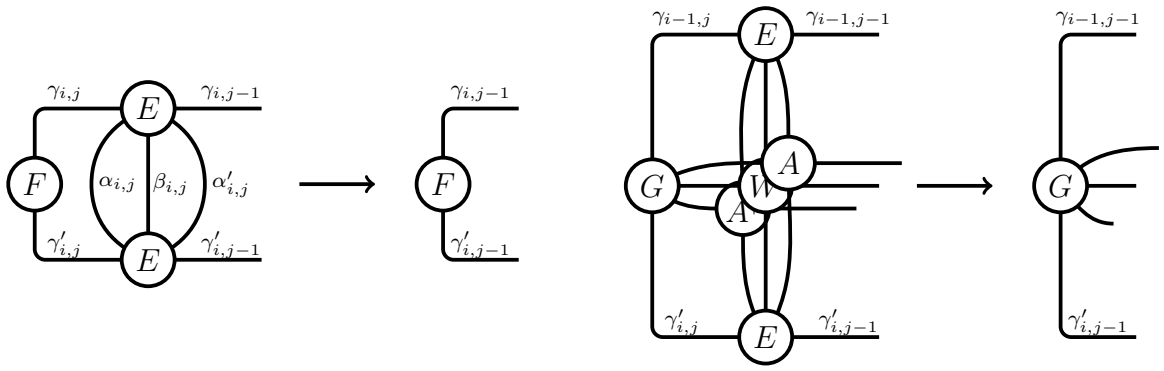
(f) Contraction.

(g) Truncated decomposition.

(h) Contraction.

Figure 6.8: Initialization of truncated environment $|\tilde{\psi}\rangle$.

indices γ are fused into the index a, whereas the same projector is used for both M and B. This is followed by another fusion of the remaining $\alpha_{i,n}$ and $\beta_{i,n}$, which are virtual indices within PEPSs and PEPOs, respectively. Finding the solution to Eq. (6.8) now



(a) Intermediate contraction steps of $\langle\tilde{\psi}|\tilde{\psi}\rangle$.

(b) Intermediate contraction steps of $\langle\tilde{\psi}|\psi\rangle$.

Figure 6.9: Intermediate contraction steps.

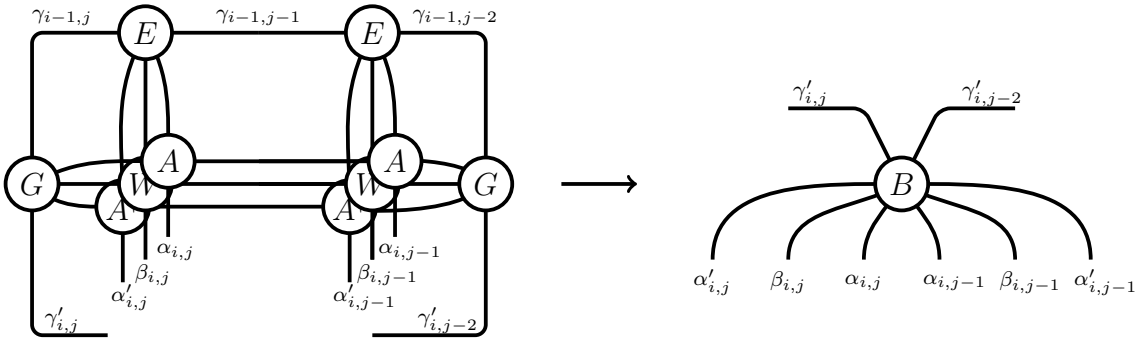


Figure 6.10: Calculation of the inhomogeneity B.

amounts to solving

$$M \cdot X = B \tag{6.9}$$

for all columns of B. Since M can, in principle, be rank-deficient, one must utilize a minimum-norm solver. The resulting X is split through the adjoint projectors, yielding the optimized, joint environment block we were looking for. Finally, X is decomposed into two single tensors, $E_{i,j}$ and $E_{i,j-1}$, generating the cumulative index $\gamma_{i,j-1}$. So, similarily to the DMRG, one can now sweep repeatedly left and right to progressively improve the environment, while the singular values of the SVD are factored into the block which points into the current sweep direction. The state $|\tilde{\psi}\rangle$ then converges to the best approximation of $|\psi\rangle$ within the given maximum bond dimension of $\gamma_{i,j}$. At the cost of accuracy, the scaling with respect to system size of contracting PEPS-based tensor networks has thus become linear, rather than exponential.

Note that the approach presented above is not the only possibility for approximating environments. The original algorithm [51, 124], upon which our scheme is based, optimizes the environment tensors individually in a single-site manner. This approach can be used in addition to the two-block optimization described above, but, on its own, it leaves the environment in the basis given by the initalization and cannot converge to the best

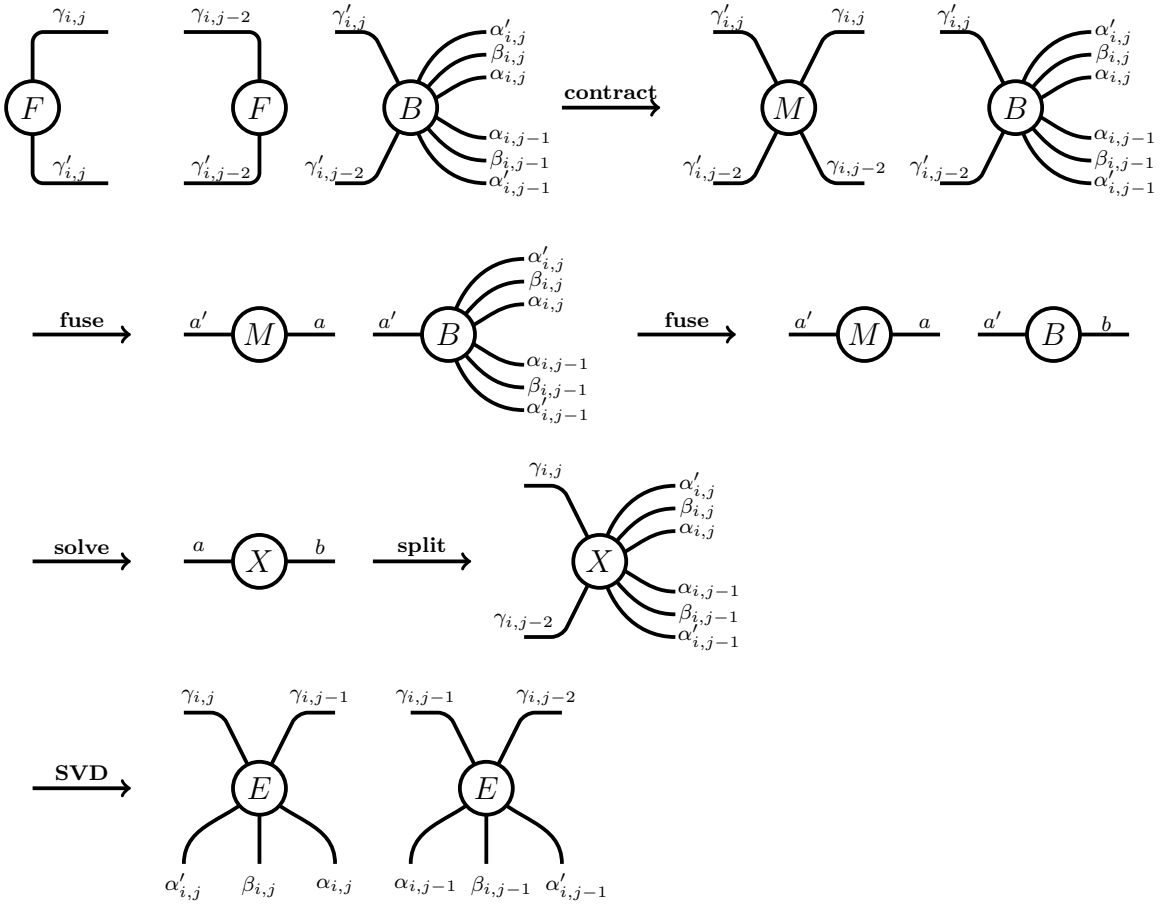


Figure 6.11: Solving for a joint environment tensor.

solution, as bonds are not optimized. In another approximation [125], physical indices instead of virtual ones are bundled, which turns out to be less accurate than the previous approach [126]. The least expensive way is the simple update [127], which depends solely on the SVD, but is also too inaccurate. Lubasch et al. [126] proposed a more elaborate approximation for the calculation of norms by considering larger clusters around the row or column one wants to process. In our case, this algorithm would be too expensive, as the PEPO in between the two PEPSs leads to a much higher scaling.

6.4 Variational optimization

With all the necessary preliminaries covered, we can now introduce the variational optimization of PEPSs. Suppose we are given a Hamiltonian in the form of a PEPO and target a certain sector of quantum numbers for which we want to find the ground state. The initialization of the algorithm takes place by converting a product state within that sector into a PEPS. For the optimization of two adjacent bulk-tensors, say $A_{x,y}$ and $A_{x+1,y}$, we first have to contract both the network of the Hamiltonian, $\langle \mathrm{PEPS} | \, \mathrm{PEPO} \, | \mathrm{PEPS} \rangle$ (Fig. 6.6(a)), and the network of the norm, $\langle \mathrm{PEPS} | \mathrm{PEPS} \rangle$, partially, by building environments

from above and below, using the scheme described in Chapter 6.3. This leaves one with four tensor sequences, two for the energy, $E_{i,y+1}$ and $E_{i,y-1}$, and two for the norm, $N_{i,y+1}$ and $N_{i,y-1}$. Afterwards, environment blocks within row y are calculated by starting at both $x = 0$ and $x = x_{\max}$ and contracting in accordance with Fig. 6.12. The tensors $A_{i,y}$,

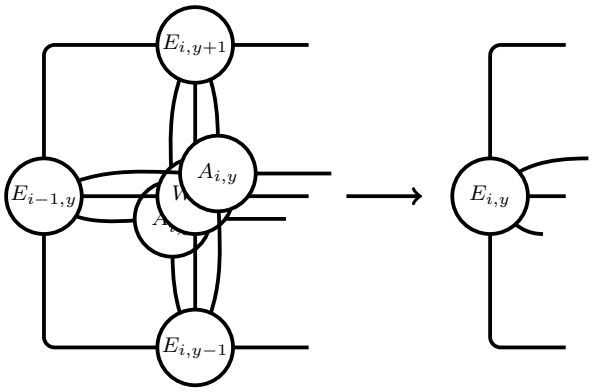
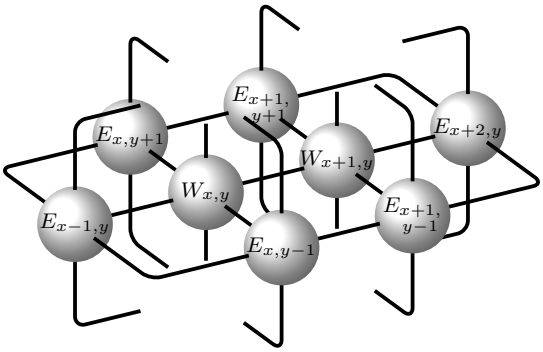


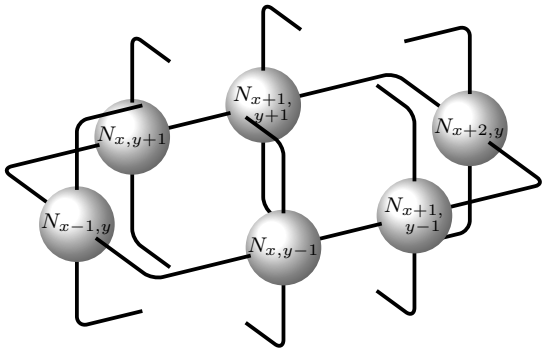
Figure 6.12: Zipper-contraction of blocks between two approximated environments.

$W_{i,y}$, and $A^\dagger$ belong to the PEPS, PEPO, and adjoint PEPS, respectively; note that the latter two are partially covered in the figure. On the left, $E_{i-1,y}$ initially does not exist at the edge of the lattice and then contains the outcome of the previous contraction for $i > 0$. The contractions on the right take place analogously. One can picture this process as two zippers closing in on the two sites (x, y) and $(x + 1, y)$.

The resulting setup is depicted in Fig. 6.13. The effective Hamiltonian H_{eff} consists of six environment tensors E and the last two tensors of the PEPO that are not contracted, $W_{x,y}$ and $W_{x+1,y}$, while the environment for the case of the norm, N_{env}, consists of just six norm tensors N. Bonds connecting two E's or two N's are cumulative and are created



(a) Effective Hamiltonian H_{eff}.



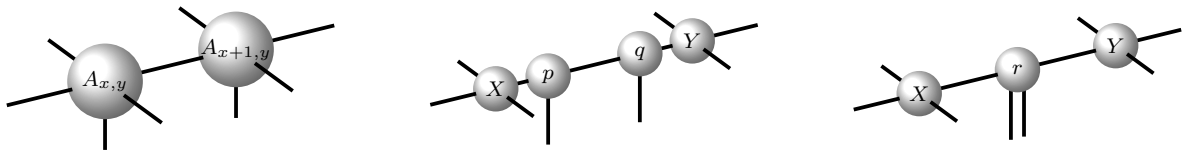
(b) Norm environment N_{env}.

Figure 6.13: Two-site optimization.

during the approximation of the environment, those connecting both W or one W and one E represent states in an FSM, the indices sticking out of E's or N's encode quantum mechanical entanglement within the PEPS, and the dangling links of both W's are physical Hilbert spaces. As a whole, the clusters H_{eff} and N_{env} determine the generalized eigenvalue

problem, Eq. (2.19), for which we want to find the optimal wave function $|\psi\rangle = A_{x,y} \cdot A_{x+1,y}$. Note that PEPSs have no equivalent to the Schmidt form, in which the norm of a single tensor is equal to the norm of the overall wave function. Otherwise, we could find a gauge so that $\langle\psi| N_{\text{env}} |\psi\rangle = \langle\psi|\psi\rangle$, as in the DMRG.

In principle, one could now proceed with an optimization of $|\psi\rangle$ as a whole through iterative diagonalization. Here, however, we split the process into the optimization of the bond between $A_{x,y}$ and $A_{x+1,y}$, and an optimization of $A_{x,y}$ and $A_{x+1,y}$ individually, which is far more efficient. The bond optimization was proposed by Corboz [128] and its initialization is illustrated in Fig. 6.14. The PEPS tensors $A_{x,y}$ and $A_{x+1,y}$ are decomposed



(a) Two adjacent PEPS tensors. (b) Decompositions. (c) Contraction.

Figure 6.14: Decomposition of PEPS-tensors.

into (X,p) and (q,Y), respectively. The middle tensors p and q are then contracted to form r, which carries both of the physical indices. The tensors X and Y are then absorbed into the environment, and r is variationally optimized with respect to $\hat{H} = H_{\text{eff}} \cdot X \cdot X^\dagger \cdot Y \cdot Y^\dagger$ and $\hat{N} = N_{\text{env}} \cdot X \cdot X^\dagger \cdot Y \cdot Y^\dagger$. Afterwards, r is factorized back into p and q, and the bond in between is truncated with respect to a predetermined maximum number of states. Due to the cyclic nature of the tensor network, the singular values are not equal to Schmidt coefficients and, consequentially, cannot be directly related to entanglement. The truncation can thus be enhanced by a full update (FU) [60, 129], also called full environment truncation (FET) [130]. The idea behind this procedure is the same as the one behind the environment approximation in Section 6.3: Given a full vector $|\psi\rangle$, the best approximation $|\tilde{\psi}\rangle$ is given by the maximal overlap $\langle\tilde{\psi}|\tilde{\psi}\rangle = \langle\tilde{\psi}|\psi\rangle$, or, in this case,

$$\hat{N}\, p^\dagger\, q^\dagger\, p\, q = \hat{N}\, p^\dagger\, q^\dagger\, r\,. \tag{6.10}$$

If one of p or q is fixed in the beginning through a truncated SVD of r, the other can the be determined by solving

$$\left(\hat{N}\, q^\dagger\, q\right)\, p = \hat{N}\, q^\dagger\, r\,, \tag{6.11}$$

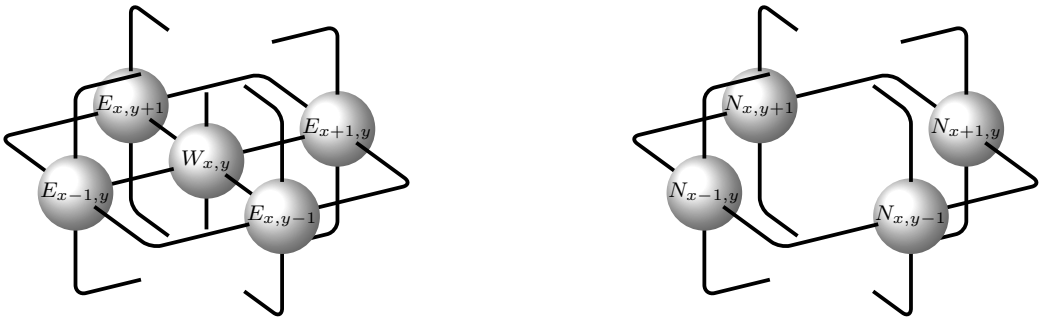
$$\left(\hat{N}\, p^\dagger\, p\right)\, q = \hat{N}\, q^\dagger\, r\,. \tag{6.12}$$

Alternating between Eq. (6.11) and (6.12) thus successively improves the results for p

and q in an iterative manner. The quality of the FU is determined by the cost function,

$$d\,(p,q) = \hat{N}\,r\,r^{\dagger} + \hat{N}\,p\,q\,p^{\dagger}\,q^{\dagger} - \hat{N}\,r\,p^{\dagger}\,q^{\dagger} - \hat{N}\,p\,q\,r^{\dagger}\,, \tag{6.13}$$

which is to a PEPS what the truncation error is to an MPS. Since the operations in Eq. (6.11) and (6.12) are negligibly expensive compared to the environment approximation and iterative diagonalization, they may be repeated until $d\,(p,q)$ converges to a constant value up to machine precision. The values of d can additionally be plotted versus the energy and extrapolated to $d = 0$ using a polynomial fit to give a better approximation to the energy [131]. After the bond optimization, we recover both $A_{x,y} = X \cdot p$ and $A_{x+1,y} = q \cdot Y$ and optimize them through alternating iterative diagonalizations on both sites, the setup of which is shown in Fig. 6.15.



(a) Effective Hamiltonian. (b) Norm environment.

Figure 6.15: One-site optimization

In summary, an optimization step is executed as follows: The bond between two adjacent tensors is isolated by forming the tensor r, which is then optimized using the Davidson algorithm. The resulting r is decomposed via a truncated SVD, and one of p or q is used as a starting point for the full update. The values of the cost function can be saved for future energy extrapolations. Both p and q are absorbed into X and Y to recover the original PEPS tensors $A_{x,y}$ and $A_{x+1,y}$. The right tensor $A_{x+1,y}$ is then locked by a zipper-contraction of the environment, i.e., according to Fig. 6.12, which allows for $A_{x,y}$ to be optimized using the Davidson algorithm in a single site manner. Afterwards, $A_{x,y}$ is locked and $A_{x+1,y}$ is optimized. One may then apply the bond optimization again and repeat the procedure.

The overall variational optimization of the whole PEPS is illustrated in Fig. 6.16 for a 4x4 lattice. The sketches are to be read rowwise and from left to right, as indicated by the numbering. Sketch 1 in the upper left corner is a tiny aerial view of Fig. 6.6(a) and represents the starting point. The red arrows in sketch 2 denote three successive constructions of approximate environments for both the energy and the norm, which isolates the lowest row. The blue arrow in sketch 3 stands for the zipper contraction of the two sites on the right, which allows for the two-site optimization of the lower-left corner

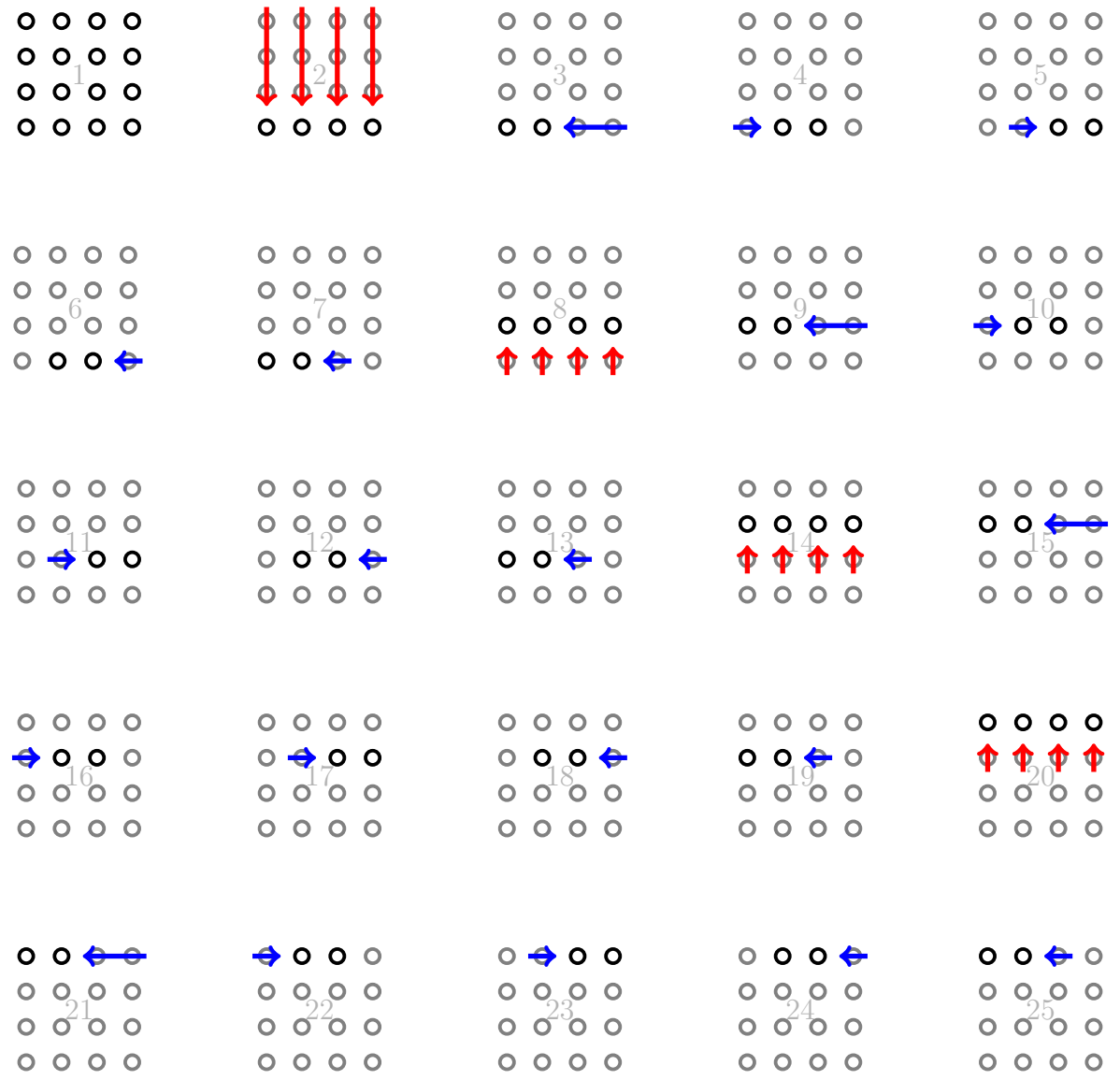


Figure 6.16: Horizontal sweep in variational optimization of PEPS on a 4x4 lattice.

by the mechanism described above. Afterwards, we move two steps to the right, optimize both times, reverse direction, and again optimize two times. In sketch 8, the treatment of the next row is prepared by building the environment of the recently processed first row. Sketches 9 to 13 illustrate the optimization of row 2 in the same order. The remaining pictures follow the same pattern, until the uppermost row is reached. As a whole, the 25 sketches illustrate one horizontal sweep, which consists of 4 DMRG-like optimizations of each row and, in the case of the Hubbard model, move charge and spin horizontally across the lattice to minimize the energy. A vertical sweep can be implemented simply by rotating every sketch in Fig. 6.16 by 90 degrees. We call a horizontal sweep followed by a vertical sweep a full sweep.

As a final remark, we note that, due to the approximate construction of environments,

the Ritz-principle is, unfortunately, violated, and the energy of the PEPS is not necessarily bounded from below by the ground state energy of the Hamiltonian. In subsequent chapters, we will discuss to what extent the variational aspect of the algorithm is distorted.

6.5 Redistribution of weight

Another problem within PEPS-based algorithms is the distribution of weight. Consider a sequence of tensors E that either constitute an approximate environment (Fig. 6.7) or surround a PEPS-tensor to be optimized (Fig. 6.15). Although the square roots of the singular values are consequently factored into both sides after decompositions, connected E's can still exhibit an overall gradient of weight that, if too large, can lead to rounding errors.

In the following, we apply two kinds of similarity transformations that smear out weight among multiple E's and stabilize the algorithm. First, we calculate the mean square x_j of each tensor E_j and iteratively equalize all the x_j with respect to the subsequent update of adjacent sites with the largest disparity:

$$x_j \, \lambda_j = \lambda_j^{-1} \, x_{j+1} \quad \Rightarrow \quad \lambda_j = \sqrt{\frac{x_{j+1}}{x_j}} \, . \tag{6.14}$$

The accumulated λ_j's between two x_j can be absorbed into a new quantity θ_j. Once the x_j are equal, diagonal matrices $\theta_j \mathbb{1}$ and $\theta_j^{-1} \mathbb{1}$ are inserted between all pairs E_j and E_{j+1}. In this way, weight is shifted across a sequence of environment tensors without changing the network as a whole. Subsequently, we perform the sequence of operations depicted in Fig. 6.17 on each pair: Both tensors are decomposed with the singular values shifted to the middle. The intermediate tensor T, which carries all of the weight, is then decomposed with $\sqrt{\Sigma}$ factored into both sides and redistributed through Givens rotations, as explained in Chapter 4.5. Both halves are then absorbed by the corresponding environment tensors.

Both redistribution mechanisms may be applied repeatedly, as their overhead is small compared to the main bottlenecks of the algorithm.

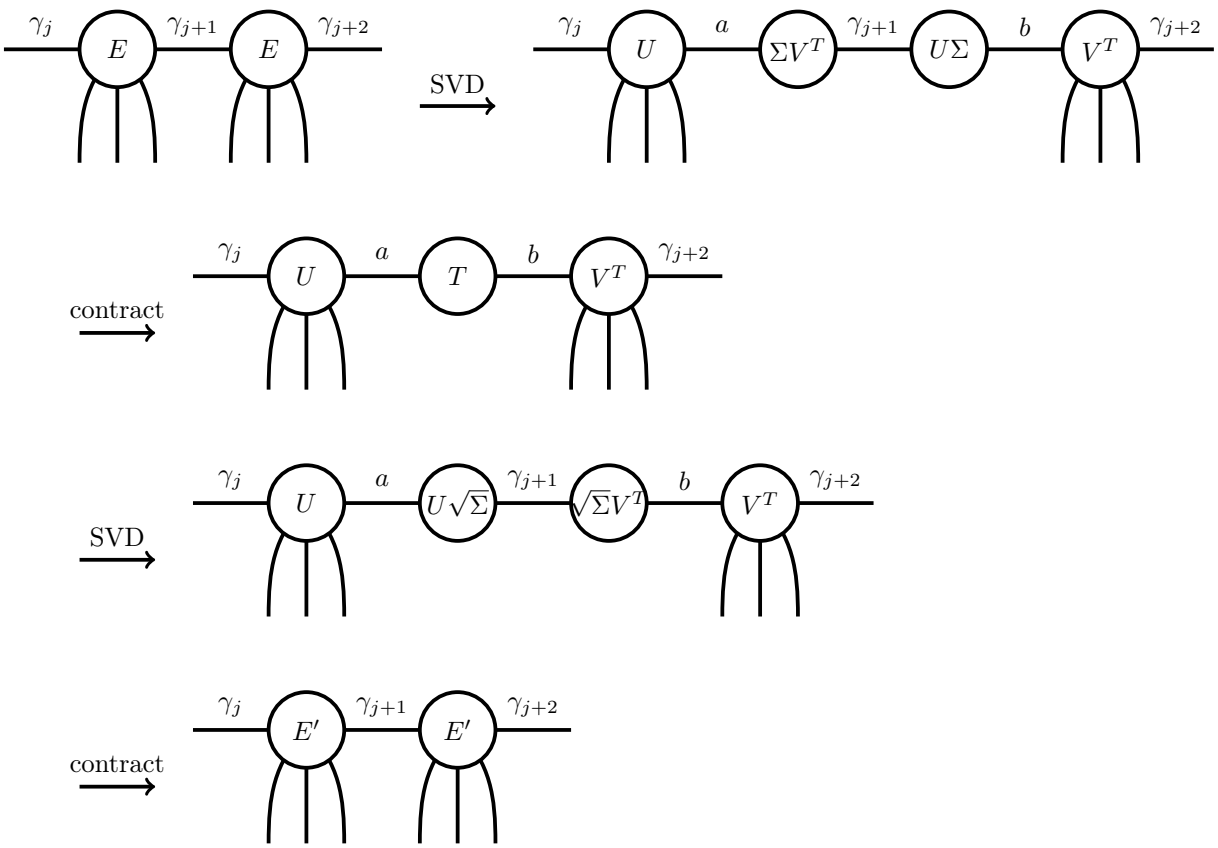


Figure 6.17: Redistribution of weight between two adjacent environment tensors.

7 Benchmark results

7.1 The parameters

Prior to the presentation of simulation results, we discuss all relevant input parameters. First, recall the Hubbard Hamiltonian:

$$H = -t \sum_{\langle i,j\rangle,\sigma} \left(c^\dagger_{i,\sigma}\, c_{j,\sigma} + \text{h.c.} \right) + U \sum_i n_{i,\uparrow}\, n_{i,\downarrow} - \mu \sum_{i,\sigma} n_{i,\sigma}\,, \qquad n_{i,\sigma} = c^\dagger_{i,\sigma}\, c_{i,\sigma}\,. \tag{7.1}$$

Hopping is restricted to nearest neighbors only, as indicated by the notation $\langle i, j\rangle$. Since the scale of the Hamiltonian is set by $t = 1$, there are two independent physical parameters, U/t and μ/t, representing onsite interaction and doping, respectively. The μ-term is usually not explicitly included in H, but implicitly taken care of by the particle number N of the initial wave function, yielding a canonical ensemble. Most implementations of tensor network algorithms incorporate abelian symmetries, i.e., $\mathrm{U}(1)_{\text{spin}} \otimes \mathrm{U}(1)_{\text{charge}}$ for the Hubbard model. Bonds within tensor networks are then parameterized by the states $|m, c_z, t\rangle$, with m representing the z-component of spin, c_z the deviation of the particle number from half-filling, and t an additional degeneracy, respectively. In the absence of a magnetic field B, one can take advantage of the full spin rotation symmetry and use the $\mathrm{SU}(2)_{\text{spin}} \otimes \mathrm{U}(1)_{\text{charge}}$–group. This yields the far richer states $|s, c_z, t\rangle$, whose total spin s covers the whole multiplet $m = -s, ..., m = s$ through intertwiners.

The central numerical parameter in tensor network algorithms is the maximum number of virtual states within the tensor network state, usually refered to as simply the bond dimension D. For Hamiltonians whose ground states exhibit an area law, this number converges in the thermodynamic limit. For projected entangled pair states, expectation values can only be calculated approximately, as illustrated in Fig. 6.7. Hence one has to analyse the influence of the maximum bond dimension χ of the cumulative indices $\gamma_{i,j}$, which is not to be confused with D. Furthermore, we need to examine the number of sweeps back and forth necessary to determine the optimal environment tensors in Fig. 6.11 and the best routine for solving Eq. (6.9). For the Davidson algorithm (Sec. 2.1), which is at the core of the variational optimization within tensor networks, we face the following dilemma: Each PEPS-tensor A_i pending optimization is part of the effective Hamiltonian of all other PEPS-tensors and vice versa. Hence, it does not pay off to set an overly strict convergence criterion for the iterative diagonalization, as sweeping across the lattice will, in time, improve all other tensors and lead to a better effective Hamiltonian for A_i. On the other hand, too few Davidson steps increase the relative overhead of the environment approximation. Here, two adjacent PEPS tensors are processed either by a single step within the setup of Fig. 6.13, with bond environments truncated to a dimension of $\chi = 10$, or three steps, which optimize the bond as shown in Fig. 6.14. Afterwards, both tensors

are always improved individually over three steps. The whole scheme is repeated three times.

7.2 Convergence of environment approximation

As an introduction to the data analysis, Fig. 7.1 displays a single sweep of the variational optimization of the Hubbard model on a 6x6 lattice using the full $\mathrm{SU}(2)_{\mathrm{spin}} \otimes \mathrm{U}(1)_{\mathrm{charge}}$ - symmetry. On the x-axis, the optimization steps enumerate every single iteration within

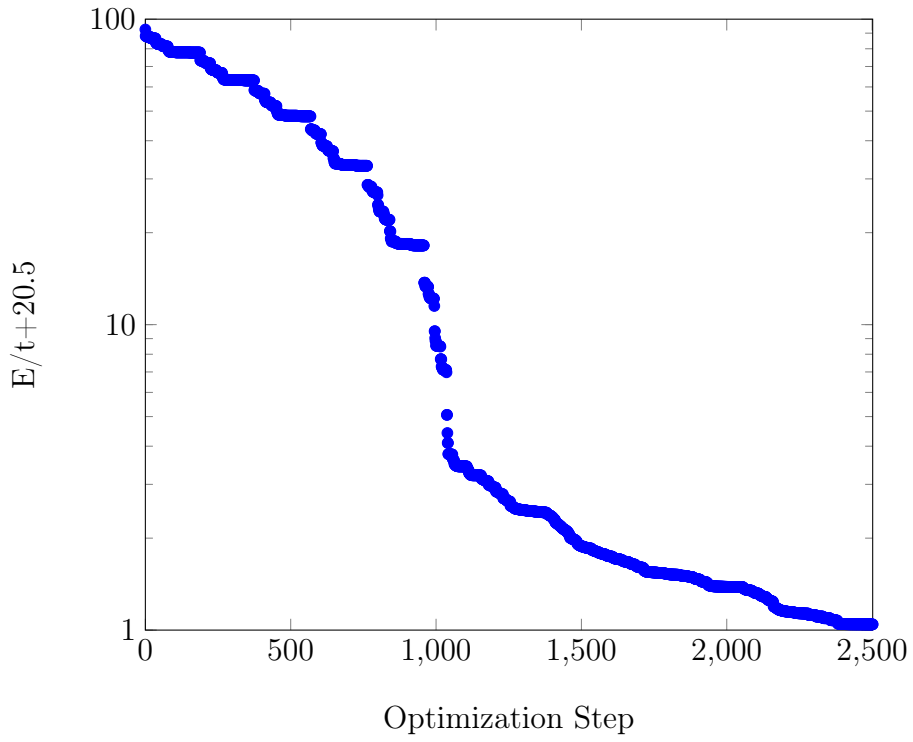


Figure 7.1: Energy convergence of first sweep for a 6x6 lattice, $U = 4$, $S = 0$, $N = 36$, $\chi = 100$, and $D = 5$.

Davidson algorithm, as one sweeps through the lattice according to the scheme depicted in Fig. 6.16. The y-axis represents the energy and has been shifted so that the lowest energies within this sweep are slightly above zero to generate a logarithmic plot. The PEPS was initialized as a product state with alternating local states $|0\rangle$ and $|\uparrow\downarrow\rangle$ to ensure half filling and a total spin of zero ($S = 0$). The bond dimension D was set to 5, and χ was set to 100. Since $U = 4$ and since half filling on 36 sites implies 18 double occupations, the algorithm starts at $E/t = 4 \cdot 18 = 72$. As already indicated in Sec. 7.1, the simulations in the current section were performed with an earlier version of the algorithm that optimizes two adjacent tensors as a whole using a truncated effective Hamiltonian and norm environment, as depicted in Fig. 6.13.

The overall trend of this one sweep verifies the variational character of the algorithm, but, before we proceed to consider more sweeps, larger systems, or other doping regimes,

we will examine the impact of the environment approximation in detail. First, we concentrate on the final 100 optimization steps of the first sweep and compare three different bond dimensions χ, executed on two different CPU models, in Fig. 7.2. While the influ-

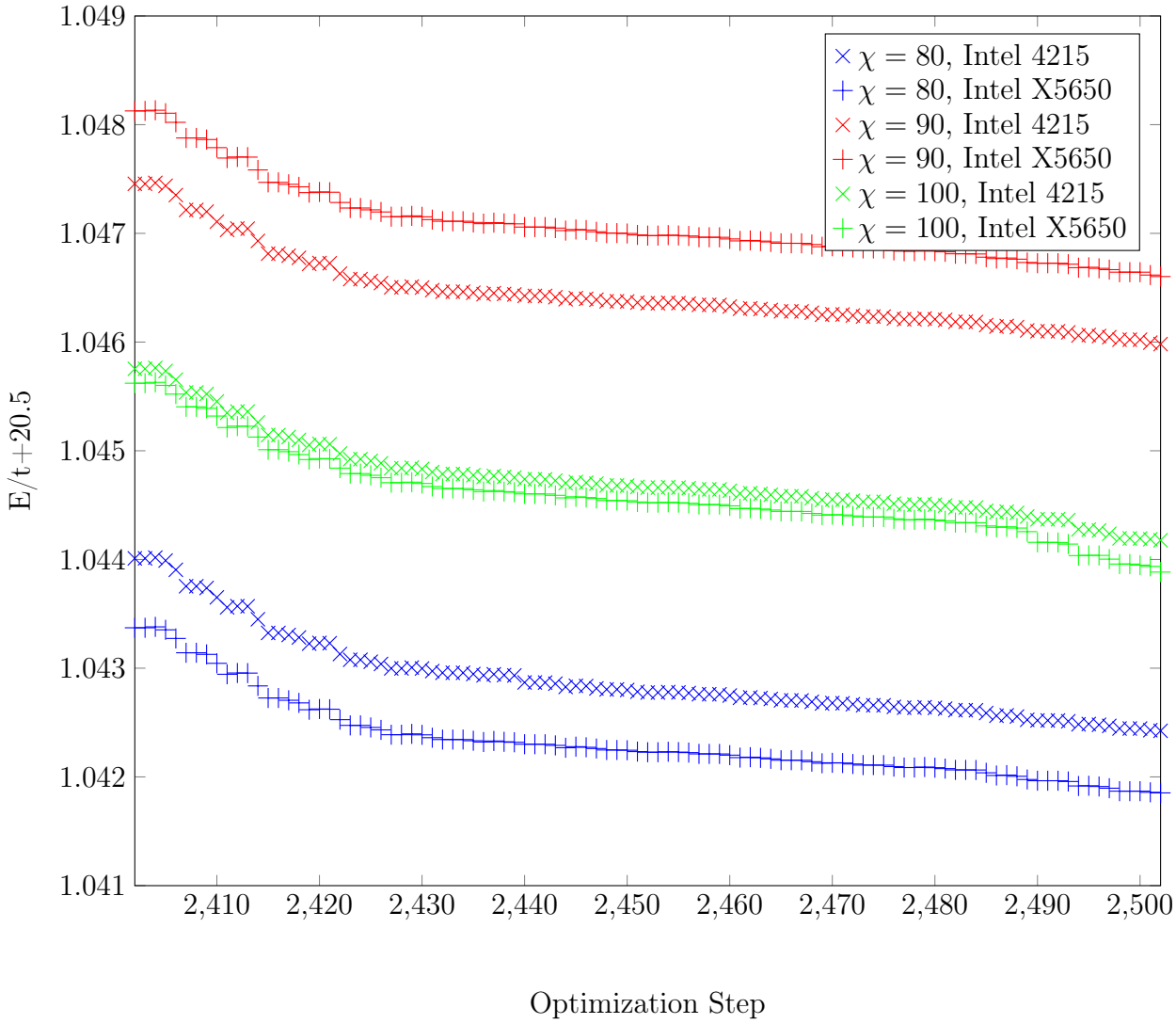


Figure 7.2: Comparison of the final steps of the energy convergence on two CPU models for a 6x6 lattice, $U = 4$, $S = 0$, $N = 36$, and $D = 5$.

ence of the bond dimension was anticipated, it is noteworthy that it is not proportional to the energy. From 80 to 90 bonds, the data points for the energy become higher, but from 90 to 100 bonds, they become lower again. Furthermore, the dependence on the CPU model is unexpected. The algorithm is deterministic and should, in principle, reproduce the same outcome to arbitrary precision for identical input parameters. The variation with respect to the underlying hardware must hence be rooted in the different implementations of low-level algebra routines, which are usually tailored to a particular CPU model. For the variational optimization of PEPSs, it is thus necessary to perform comparative convergence analyses using the same CPU type to avoid systematic error.

We now go one step further and quantify the error due to the environment approximation discussed in Sec. 6.3. Suppose the first full sweep has just finished, yielding a PEPS with fully entangled states along all horizontal and vertical bonds. The preparation of the second sweep, illustrated by the red arrows in sketch 2 of Fig. 6.16, thus reveals to what extent the environment approximation leads to a violation of the variational nature of our optimization and the calculation of observables $\langle\psi| O |\psi\rangle$ in general. If the last energy of the first sweep is $E_{\text{last},1}$, and the first energy of the second sweep is $E_{\text{first},2}$, the error can be quantified as

$$\varepsilon = \left| \frac{E_{\text{last},1} - E_{\text{first},2}}{E_{\text{last},1}} \right| . \tag{7.2}$$

Table 7.1 compares $E_{\text{last},1}$ and ε for different numbers of complete environment sweeps, i.e., sweeping back and forth, as described in Sec. 6.3, which should not be confused with the sweeps of the PEPS depicted in Fig. 6.16. In particular, an environment is first

Table 7.1: Last energy of first sweep $E_{\text{last},1}$ / energy error ε for a 6x6 lattice, $U = 4$, $S = 0$, $N = 36$, and $D = 5$.

sweeps \ bonds χ	80	90	100
$1, 0$	–	–	–
$1, 1$	–	–	–
$1, 2$	$-19.5365/3.74497 \cdot 10^{-4}$	$-19.5209/1.76979 \cdot 10^{-4}$	$-19.5266/1.02936 \cdot 10^{-4}$
$1, 3$	$-19.7531/1.0183 \cdot 10^{-5}$	$-19.7540/2.695 \cdot 10^{-6}$	$-19.7543/3.0149 \cdot 10^{-5}$
$2, 0$	$-19.4523/2.8593 \cdot 10^{-5}$	$-19.4529/3.5193 \cdot 10^{-5}$	$-19.4542/1.1621 \cdot 10^{-5}$
$2, 1$	$-19.4587/1.8211 \cdot 10^{-5}$	$-19.4534/8.770 \cdot 10^{-6}$	$-19.4518/1.648 \cdot 10^{-6}$
$2, 2$	$-19.4576/1.1167 \cdot 10^{-5}$	$-19.4540/9.288 \cdot 10^{-6}$	$-19.4558/2.641 \cdot 10^{-6}$
$2, 3$	$-19.4578/1.1188 \cdot 10^{-5}$	$-19.4533/8.117 \cdot 10^{-6}$	$-19.4561/3.036 \cdot 10^{-6}$
$3, 0$	$-19.4557/5.3497 \cdot 10^{-5}$	$-19.4520/9.485 \cdot 10^{-6}$	$-19.4540/2.615 \cdot 10^{-6}$
$3, 1$	$-19.4596/9.507 \cdot 10^{-6}$	$-19.4543/8.837 \cdot 10^{-6}$	$-19.4554/2.849 \cdot 10^{-6}$
$3, 2$	$-19.4578/1.0410 \cdot 10^{-5}$	$-19.4513/6.663 \cdot 10^{-6}$	$-19.4549/2.873 \cdot 10^{-6}$
$3, 3$	$-19.4576/1.0770 \cdot 10^{-5}$	$-19.4517/5.532 \cdot 10^{-6}$	$-19.4537/3.3359 \cdot 10^{-6}$

optimized by carrying out a number of sweeps that solve for joint-environment tensors (N_J), as depicted in Fig. 6.11, followed by further sweeps that determine the optimal blocks in a single-site manner (N_S), corresponding to the original algorithm proposed by Verstraete et al. [51]. The left column lists all cases examined in terms of (N_J,N_S) and data are shown for three different bond dimensions χ. We immediately see that carrying out just one sweep for the joint-block case is insufficient. For no or for just one following single-block sweep, the algorithm simply terminates before it can even finish the first

PEPS sweep. For two or three single-block sweeps, it manages to reach the second PEPS sweep, but $E_{\text{last},1}$ deviates too much from the value of approximately -19.45, obtained by all simulations in the lower rows. Carrying out three rather than two sweeps for joint blocks, on the other hand, does not improve the results noticably. Within the set of $N_J = 2$ sweeps, one further single-block sweep decreases ε significantly, while additional single-block sweeps are superfluous. Given a 6x6 lattice with 5 virtual states D and a bond dimension of $\chi = 90$, it is thus optimal to choose $N_J = 2$ and $N_S = 1$.

Table 7.2 shows the error in the ground state energy for various bond dimensions χ and reveals a trend that agrees with expectations: ε decreases with increasing bond dimension, and the contraction forming $\langle \text{PEPS}| \, \text{PEPO} \, |\text{PEPS}\rangle$ becomes more accurate. In addition,

Table 7.2: Energy errors ε for varying bond dimensions for a 6x6 lattice, $U = 4$, $S = 0$, $N = 36$, and $D = 5$.

χ	energy error
70	$2.585296 \cdot 10^{-5}$
80	$2.971941 \cdot 10^{-5}$
90	$7.99187 \cdot 10^{-6}$
100	$2.13387 \cdot 10^{-6}$
110	$9.4119 \cdot 10^{-7}$
120	$8.529 \cdot 10^{-8}$
130	$1.6100 \cdot 10^{-7}$
140	$1.7562 \cdot 10^{-7}$

ε seems to converge to a value of around 10^{-7}, which is well above the machine precision of approximately one in 10^{16} for double precision numbers. It is an open question as to whether this indeterminacy is inherent to the framework presented in this thesis, or is due to rounding errors of the low-level BLAS routines.

Finally, we compare the efficiency of two different minimized-norm solvers for solving Eq. (6.9) in Table 7.3. The routines are from the Linear Algebra Package (LAPACK); one is "gelsd", which is based on the SVD, the other is "gelsy", based on orthogonal factorization. The relative errors in the calculated energies are comparable with deviations

Table 7.3: Energy errors ε for varying solvers for a 6x6 lattice, $U = 4$, $S = 0$, $N = 36$, and $D = 5$.

solver \ bonds	80	90	100
gelsd	$1.5782 \cdot 10^{-5}$	$9.507 \cdot 10^{-6}$	$2.442 \cdot 10^{-6}$
gelsy	$1.1857 \cdot 10^{-5}$	$9.348 \cdot 10^{-6}$	$2.816 \cdot 10^{-6}$

within $5 \cdot 10^{-6}$. The solver gelsy, however, is distinctly faster.

Finally, we discuss the statistical noise in both the energy and in its error. In Table 7.1,

in particular, it can be seen that $E_{\text{last},1}$ oscillates in the third and fourth decimal places for multiple sweeps and bond dimensions. While the order of magnitude of ε follows a consistent trend for varying numerical parameters, the prefactor varies stochastically. As already indicated, we take $N_J = 2$, $N_S = 1$ and use gelsy as a solver from now on. The bond dimension χ of the following simulations is chosen so that a stable progression of the energy and a tolerable calculation time occurs.

7.3 Convergence of ground state

In this section, we push the optimization further and explore to what extent the ground state of the Hubbard model can actually be determined using fPEPS. While the truncated two-site optimization used Section 7.2 is sufficient to generate a fully entangled PEPS that can then be utilized to examine the influence of the environment approximation with one sweep, it turns out to be inadequate for an actual variational optimization over multiple sweeps, as the energy begins to oscillate erratically in subsequent sweeps. The bond-optimization variant, Fig. 6.14, provides better convergence at first, but triggers another problem within the implementation of the SU (2)–symmetry: The calculation of Eq. (4.32), which is necessary for the contraction of two SU (2)–invariant tensors, becomes an unbearable bottleneck, because much more intricate spin combinations must be calculated. Coming up with a more efficient implementation exceeds the scope of this thesis, which is why, in the following simulations, we incorporate only the $\mathrm{U}(1)_{\text{spin}} \otimes \mathrm{U}(1)_{\text{charge}}$–symmetry group.

In Fig. 7.3, the energy convergence on a 3x3 lattice is examined for $U = 4$, 1/9 doping, and $S_z = 0$. Four sweeps were carried out with a bond dimension χ of 300 and various numbers of states D. The abscissa ranges over every individual optimization step within the Davidson iterations, while the relative error of the energy is the ordinate. The reference value of the ground state energy, $E_0 \approx -6.8216$, was determined using a DMRG algorithm with 14 sweeps and 256 states, which exhausts the whole Hilbert space and is thus numerically exact.

Using a higher bond dimension and more virtual states reveals an unfortunate aspect of the fPEPS-algorithm – its computational cost. While both exact diagonalization (ED) and DMRG determine ground states on 3x3 lattices exactly using a few seconds of CPU time and a negligible amount of memory, fPEPS with 12 states reached the computational limit of our computer cluster. The simulation took about two days, needed up to 165 GB for intermediate tensor contractions and yet yields an energy that is almost 3% above E_0. The main obstacle in terms of memory usage is the contraction of the inhomogeneity, Fig. 6.10, in particular, the calculation of one–half of the network. For larger systems, the calculation time is dominated by the permutation of indices in Eq. (4.25), which is necessary to prepare tensors for standard matrix routines.

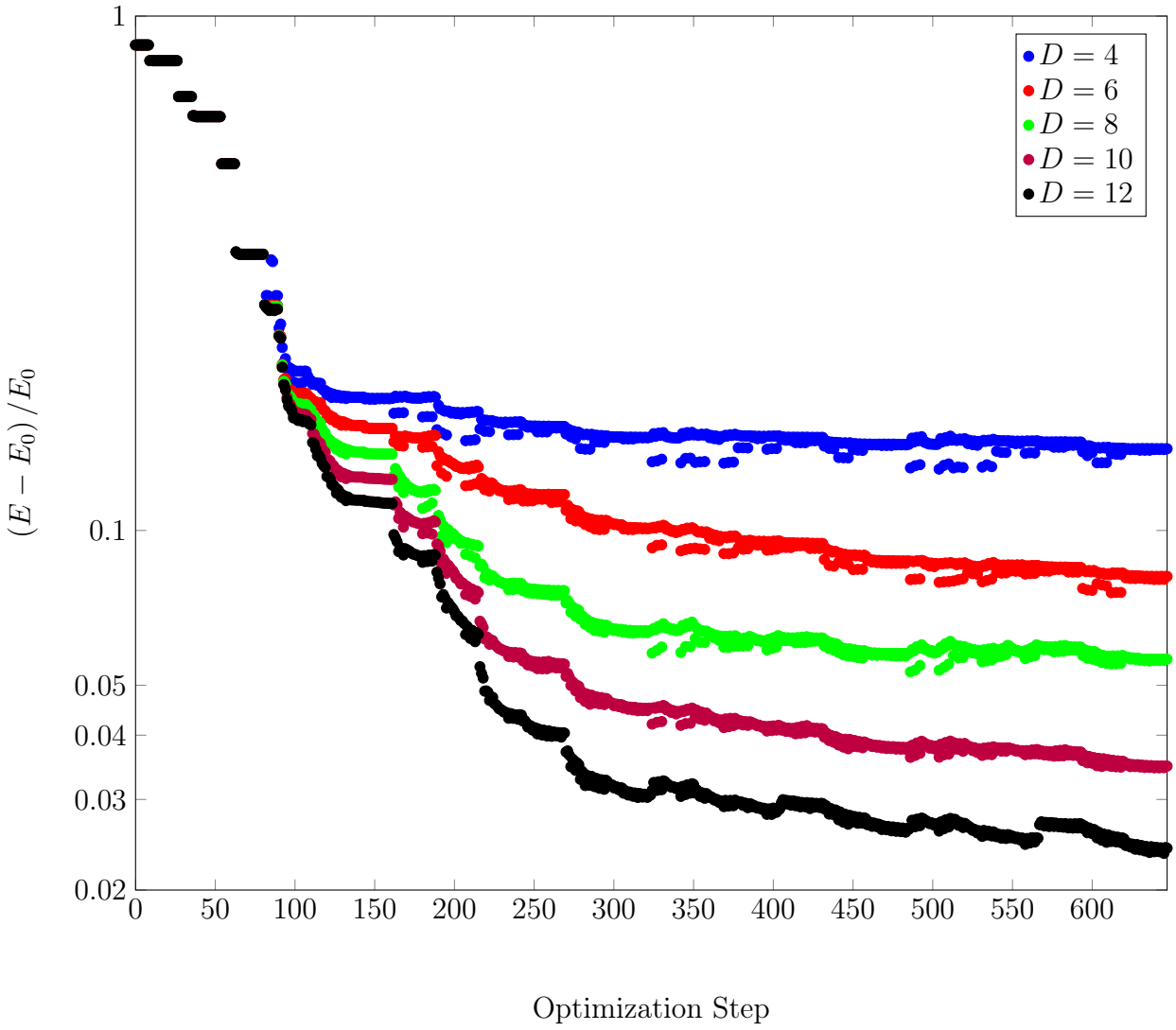


Figure 7.3: Energy convergence on a 3x3 lattice for 4 sweeps, $U = 4$, $S_z = 0$, $N = 8$, and $\chi = 300$.

Having established that the algorithm still requires much further optimization to reproduce the exact ground state for small systems, we also note that the energy at least approaches E_0 as the number of states is increased. Given the setup in Fig. 7.3, one can estimate the necessary number of states to reproduce E_0 to be around 16, which is, in fact, the maximum number used in the iPEPS calculations of the Hubbard model in Ref. [64], which is specified in Ref. [132].

The data points that appear to be outliers below the curves can be explained as follows: During a bond optimization, new virtual states are generated between two physical Hilbert spaces. The subsequent decomposition must be truncated to a predetermined maximum number of states to prevent the tensor network from growing exponentially. This inevitably strips entanglement from the wavefunction, bringing the energy back to a higher level. One can see how, for 4 states, this behavior is the most distinct, while it decreases as the number of states is increased and is barely visible for 12 states.

The discontinuities of the curve corresponding to 12 states are due to the environment approximation, which becomes more inaccurate as the number of states D increases with a constant bond dimension χ. The highest relative error in the energy [Eq. (7.2)] of $\epsilon \approx 0.002134$ appears in the preparation phase for the fourth vertical sweep, visible as the upward jump between steps 550 and 600 in Fig. 7.3. This shows that a bond dimension χ of 300 is barely sufficient for 12 states and needs to be increased if more entanglement is to be handled.

At this point, we can already conclude that PEPS-based algorithms are very inefficient for systems within the feasibility of ED and DMRG. While more effort needs to be put into testing the convergence on small lattices, the true potential of the fPEPS algorithm lies in treating larger systems, as the computational effort increases exponentially for ED and DMRG, but only linearly for PEPSs. With the current code, larger systems can only be treated if the number of states is kept modest. In Fig. 7.4, we examine an 8x8 lattice with 5 states and 100 bonds for U=8 and 1/8 doping, which is a controversial point in the ground-state phase diagram of the Hubbard model. Zheng et al. [64] have carried out

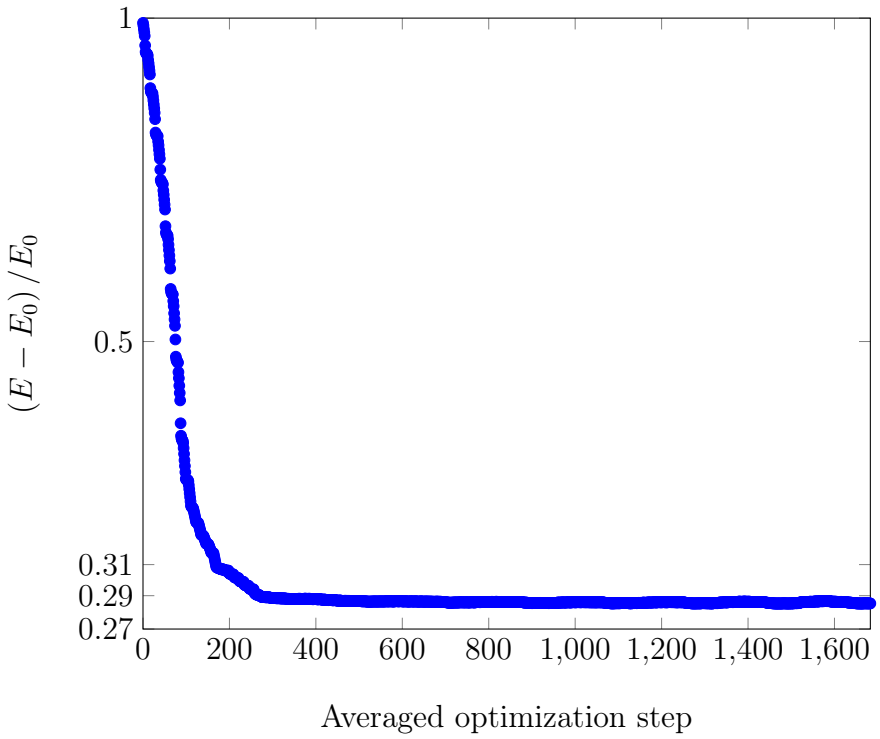


Figure 7.4: Energy convergence on a 8x8 lattice for 9 sweeps, U=8, $S_z = 0$, 1/8 doping, $\chi = 100$, $D = 5$.

various simulations with different system sizes and boundary conditions for these physical parameters, and provided averaged values for the ground-state energy per site, E_0/N, in the interval $[-0.767, -0.762]$. Here, we thus use $E_0/N \approx -0.765$ as a reference value to estimate the relative error of the fPEPS algorithm. Due to the larger number of sweeps, the energy points displayed are the final energies of each Davidson iteration, averaged

over 10 consecutive optimization steps. The energy converges to a value far above E_0, which is expected due to the small number of states and the insights gained from Fig. 7.3. However, the convergence is smooth and the energy has a stable value in subsequent sweeps, verifying the stability of the algorithm.

The benign behaviour of the energy is reflected in the progression of the charge and spin densities, depicted in Fig. 7.5. Red arrows, blue arrows, and green circles represent down-spins, up-spins, and holes, respectively. Black numbers above the diagram are averaged absolute values of the spins in the column below, i.e.,

$$|\bar{S}| = \frac{1}{8} \sum_{i=1}^{8} \frac{1}{2} \left| \hat{n}_{i,\uparrow} - \hat{n}_{i,\downarrow} \right| . \tag{7.3}$$

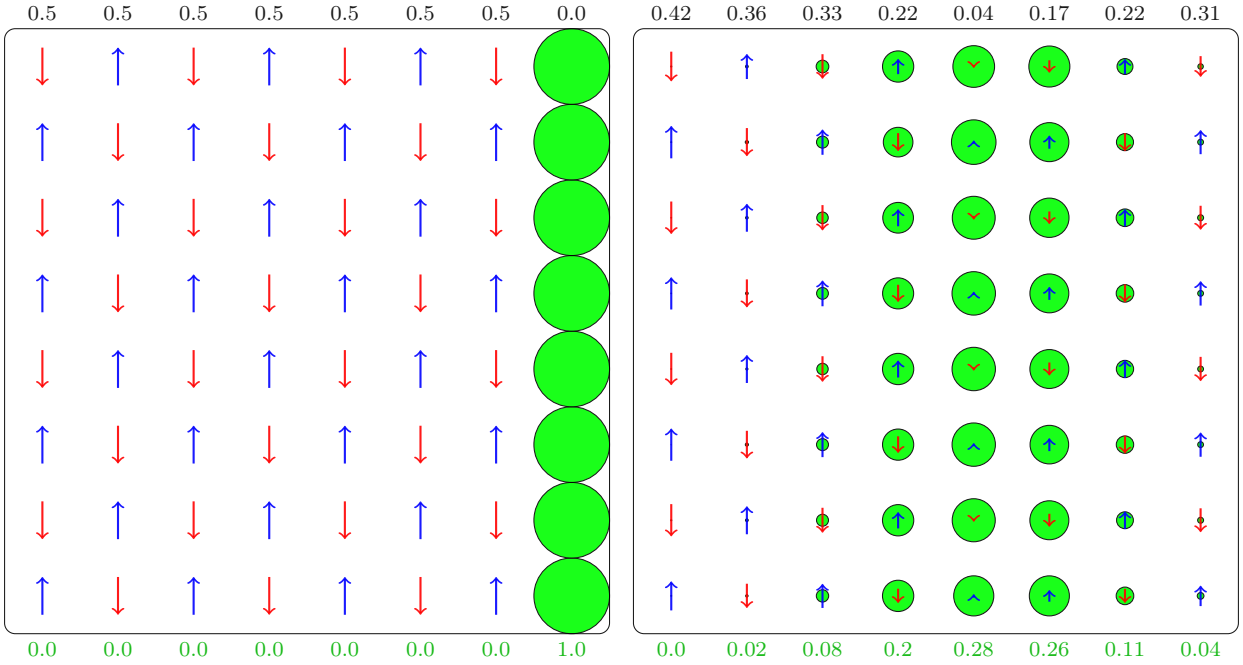
The green numbers below the bottom row are averaged hole densities,

$$\bar{n} = \frac{1}{8} \sum_{i=1}^{8} \left(1 - \hat{n}_{i,\uparrow} - \hat{n}_{i,\downarrow} \right) , \tag{7.4}$$

for the corresponding column. Fig. 7.5(a) shows the configuration of the initial product state. To implement 1/8 doping, the rightmost column has only holes, while all other sites contain one electron each with alternating spins. After one sweep, the holes are delocalized and moved to the middle, as shown in Fig. 7.5(b). The second sweep in Fig. 7.5(c) brings the configuration of charge and spin closer to a symmetric arrangement. After 9 sweeps, Fig. 7.5(d), the values change only marginaly, which corresponds to the convergence of the energy in Fig. 7.4. We conclude that, although 5 states are far too few to reach the actual ground state energy, they are already enough to qualitatively reproduce stripe structures of oscillating charge and spin densities.

A slight increase of the number of states even brings the numerical values of obervables within the range predicted by Zheng et al. [64]. Fig. 7.6 shows two curves, one for 150 bonds and 7 states, the other for 200 bonds and 8 states. To accelerate convergence, the first horizontal sweeps were conducted 5 times to spread out the holes as soon as possible, as reflected in the 8 plateaus. Subsequently, the simulation with 7 states carried out one vertical sweep and another ordinary full sweep. The calculation for 8 states was terminated after one subsequent vertical sweep. In both cases, the final energies of the Davidson iterations were averaged over 5 optimization steps.

The densities corresponding to 7 states are illustrated in Fig. 7.7. The numerical values are marginally decreased compared to Fig. 7.5, but still quite asymmetric. Fig. 7.8, which shows the charge and spin configurations for 8 states and after only five consecutive horizontal and one vertical sweep, finally resembles a symmetric distribution, except for the spins at the left and right columns. Compared to Zheng et al. [64], the numerical values come closest to the result of the density matrix embedding theory.



(a) Initial state.

(b) Sweep 1.

(c) Sweep 2.

(d) Sweep 9.

Figure 7.5: Progression of charge and spin densities during optimization for a 8x8 lattice, $D = 5$, $\chi = 100$, $U = 8$, $S_z = 0$, and 1/8 filling.

At this point, we have reached the computational limit of the current version of the algorithm. Simulations with more states, bonds, and sweeps are projected to take several weeks of clock time on our cluster for an 8x8 lattice. Although the variational optimization of finite PEPSs is not yet at the stage to deliver competitive ground state energies due

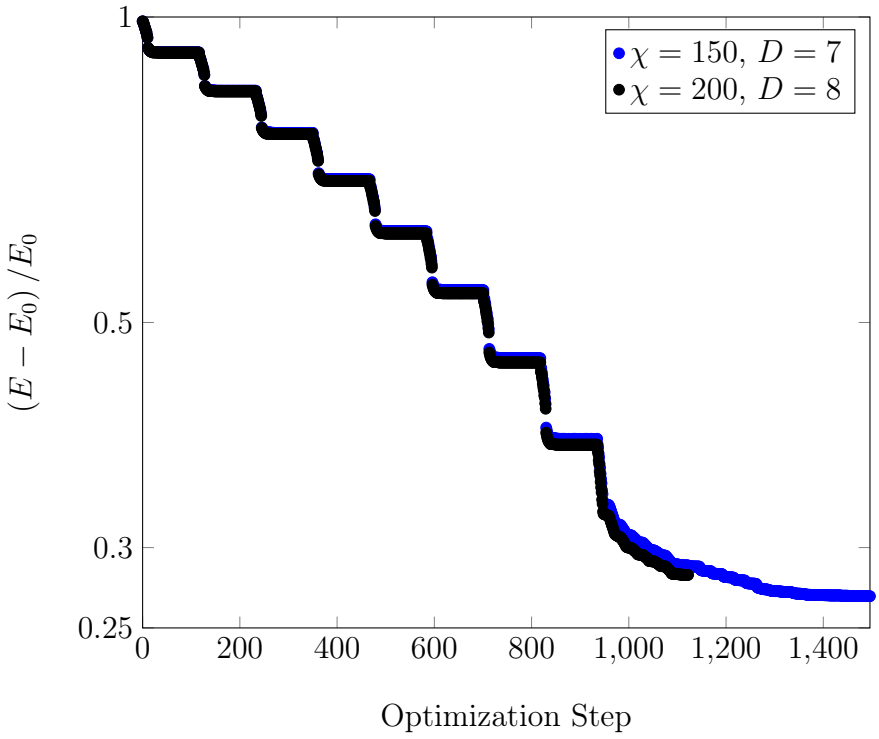
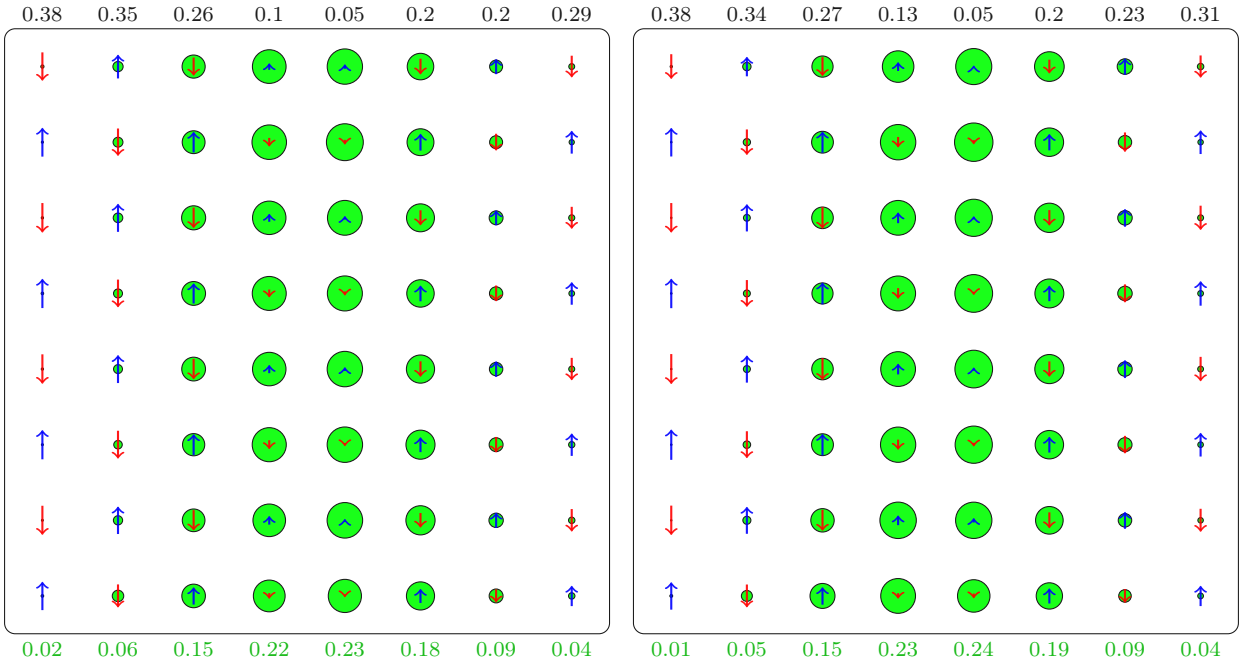


Figure 7.6: Energy convergence on a 8x8 lattice, $U = 8$, $S_z = 0$, 1/8 doping.



(a) Sweep 1.

(b) Sweep 2.

Figure 7.7: Charge and spin density for $\chi = 150$, $D = 7$, $U = 8$, $S_z = 0$, and 1/8 filling.

to several numerical bottlenecks, the development appears to be moving in the right direction.

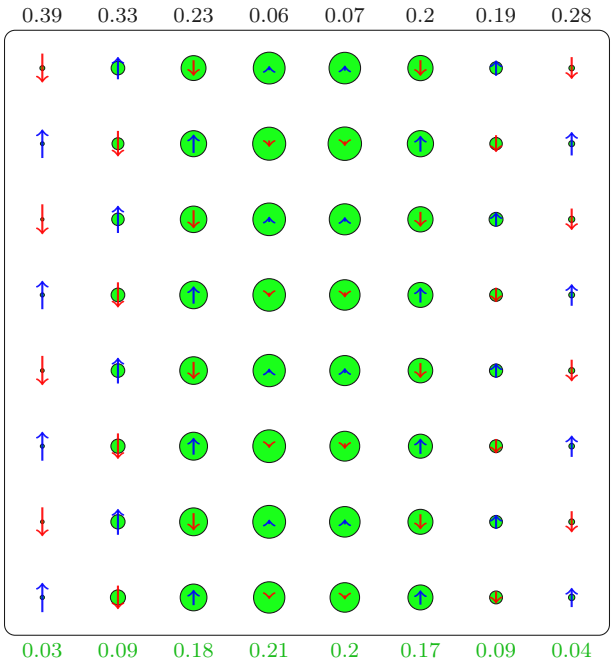


Figure 7.8: Charge and spin density for $\chi = 200$, $D = 8$, $U = 8$, $S_z = 0$, 1/8 filling, and one sweep.

8 Conclusion

This thesis developed the groundwork to carry out variational optimization of finite projected entangled pair states. The essential building blocks developed are an efficient mechanism contracting multiple block-sparse tensors (Sec. 4.6), an algorithm to create projected entangled pair operators for arbitrary local Hamiltonians (Sec. 6.2), a new procedure which optimizes the environment within PEPS-based contractions using a two-block configuration (Sec. 6.3), and a generic scheme to incorporate non-abelian symmetries, in particular $\mathrm{SU}(2)_{\mathrm{spin}}$ symmetry (Chapter 4). Together with a few minor tricks and improvements, these four major technical contributions were incorporated into the existing body of knowledge about tensor networks to create one comprehensive algorithm for the ground state calculation of two-dimensional quantum systems, which was implemented in the C++ programming language. Given a Hamiltonian which follows the entropy area law, meaning a ground state whose von-Neumann entropy is bounded from above, projected entangled pair states offer a representation which is tailored to this particular feature and allows one to variationally optimize the wavefunction locally. By construction, the optimization of PEPSs scales linearly with the system size in two dimensions, which makes this algorithm a promising candidate for treating significantly larger two-dimensional systems.

The aforementioned approximate construction of the environment is a central numerical limitation of the fPEPS algorithm, which is why its impact was examined in detail in Sec. 7.2 for the Hubbard model. The optimal approach was determined to be to carry out two sweeps that optimize two adjacent environment tensors and the bonds in between, followed by a single one-block sweep which yields a slight additional improvement. The number of environment bonds has to be adapted to a particular number of states within the PEPS. A high number may yield a very accurate environment approximation, but also an overly expensive calculation time. Lower values accelerate the simulation, but increase the energy error. Approximations that are too rough, either due to too low a dimension, or too few sweeps, lead to a sudden collapse of the convergence. In this case, the energy, which seemingly converges to a finite value up to a certain point, then abruptly diverges towards minus infinity. This is an advantageous feature of fPEPS: An optimization is either distinctly stable or crashes obviously, with no shades of gray in between.

From a physical perspective, the domain of interest is the point within the phase diagram that corresponds to strong coupling and near-half filling. This parameter set is known both for its numerical difficulty and spatial inhomogeneity which is also found experimentally in high-temperature superconductors. Our calculations in this regime have exposed a few remaining numerical bottlenecks in the current framework, which will have to be dealt with in the future. First, the permutation of indices in Eq. (4.25) dominates

the calculation time for a larger bond dimension. This permutation was implemented in a straightforward fashion using nested loops and, presumably, has to be replaced by a more sophisticated scheme, for instance, by utilizing standard algebra routines which optimize the usage of CPU cache memory. The memory use is dominated by the approximation of the environment and, in particular, by the calculation of the inhomogeneity as depicted in Fig. 6.10. In some way, one should thus divide up this operation into multiple pieces to keep the threshold as low as possible. Finally, the implementation of the SU(2) symmetry, though formally correct, turns out to be too inefficent in its current form. In particular, the calculation of Eq. (4.32), which is necessary for the contraction of compactified SU(2)-tensors, dominates the calculation time excessively and had to be removed to obtain the final benchmark results. To accelerate the calculation of Eq. (4.32), one could design a sophisticated caching mechanism that stores intermediate contractions of Clebsch Gordon coefficients to be reloaded later. However, if all attempts at optimizing this operation fail, one could instead create one large database of overlaps Y in Eq. (4.32) once in advance and then use it in subsequent simulations. Either way, once SU(2) is implemented efficiently, we can expect a significant increase in numerical efficiency because states within the tensor network cover whole spin multiplets.

While our optimized version of fPEPSs does not yet deliver competitive ground state energies, Sec. 7.3 certainly verifies the overall validity of the framework. For a 3x3 lattice, the energies clearly approach the reference value given by the DMRG as the number of states is increased, while, for larger systems, the algorithm is at least stable and already produces charge and spin densities that resemble the results given in Zheng et al. [64]. Once the technical difficulties mentioned above are overcome and fPEPS hopefully evolves to provide reliable ground-state energies, it will be able to make meaningful contributions to the analysis of two-dimensional quantum systems. Its immediate application would be the comparison between stripes of different wavelength and orientation. Existing algorithms are limited by either a small unit cell size or narrow lattice widths. Finite PEPSs have the potential to treat system sizes that are larger in both directions and would thus be able to examine stripes structures that have been previously inaccessible. The next step could be the implementation of translation invariance and thus periodic boundary conditions by connecting the opposite edges of the PEPS to form a torus. This version could then become an alternative to the iPEPS algorithm [128]. Other future possibilities are the extension of the Hubbard model to involve next-nearest neighbor hopping, which is a precondition for d-wave superconductivity, including multiple energy bands, such as those in the Emery model [133], other lattice geometries, such as triangular, honeycomb and kagome, incorporating time evolution, and many more.

Acronyms

BLAS	Basic Linear Algebra Package
CGC	Clebsch-Gordan coefficient
DMRG	density-matrix renormalization group
ED	exact diagonalization
FET	full environment truncation
fPEPS	finite PEPS
FSM	finite-state machine
FU	full update
iPEPS	infinite PEPS
LAPACK	Linear Algebra Package
MPO	matrix product operator
MPS	matrix product state
PEPO	projected entangled pair operator
PEPS	projected entangled pair state
QMC	Quantum Monte Carlo
RG	renormalization group
RVB	resonating valence bond
SVD	singular value decompostion

TNS	tensor network state
TTNS	tree tensor network state

References

[1] N. F. Mott, Proc. Phys. Soc. London, Sect. A **62**, 416 (1949), ISSN 0370-1298.

[2] N. F. Mott, Rev. Mod. Phys. **40**, 677 (1968), ISSN 1539-0756.

[3] M. C. Gutzwiller, Phys. Rev. Lett. **10**, 159 (1963), ISSN 1079-7114.

[4] J. Hubbard, Proc. R. Soc. A **276**, 238 (1963), ISSN 2053-9169.

[5] J. Kanamori, Prog. Theor. Phys. **30**, 275 (1963), ISSN 0033-068X.

[6] M. C. Gutzwiller, Phys. Rev. **137**, A1726 (1965), ISSN 1536-6065.

[7] Y. Nagaoka, Phys. Rev. **147**, 392 (1966), ISSN 1536-6065.

[8] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964), ISSN 1536-6065.

[9] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965), ISSN 1536-6065.

[10] E. H. Lieb and F. Y. Wu, Phys. Rev. Lett. **20**, 1445 (1968), ISSN 1079-7114.

[11] H. Bethe, Z. Angew. Phys. **71**, 205 (1931).

[12] B. S. Shastry, Phys. Rev. Lett. **56**, 2453 (1986), ISSN 1079-7114.

[13] W. F. Brinkman and T. M. Rice, Phys. Rev. B **2**, 4302 (1970), ISSN 2469-9969.

[14] A. A. Ovchinnikov, Sov. Phys. JETP **30** (1970).

[15] W. D. Langer and D. C. Mattis, Phys. Lett. A **36**, 139 (1971), ISSN 0375-9601.

[16] H. Shiba, Phys. Rev. B **6**, 930 (1972), ISSN 2469-9969.

[17] G. Beni, T. Holstein, and P. Pincus, Phys. Rev. B **8**, 312 (1973), ISSN 2469-9969.

[18] M. Takahashi, Prog. Theor. Phys. **52**, 103 (1974), ISSN 0033-068X.

[19] V. J. Emery, Phys. Rev. B **14**, 2989 (1976), ISSN 2469-9969.

[20] J. Bardeen, L. N. Cooper, and J. R. Schrieffer, Phys. Rev. **106**, 162 (1957), ISSN 1536-6065.

[21] K. G. Wilson, Rev. Mod. Phys. **47**, 773 (1975), ISSN 1539-0756.

[22] J. E. Hirsch, Phys. Rev. B **22**, 5259 (1980), ISSN 2469-9969.

[23] J. E. Hirsch, Phys. Rev. B **31**, 4403 (1985), ISSN 2469-9969.

[24] J. G. Bednorz and K. A. Müller, Z. Phys. B: Condens. Matter **64**, 189 (1986), ISSN 1431-584X.

[25] P. W. Anderson, G. Baskaran, Z. Zou, and T. Hsu, Phys. Rev. Lett. **58**, 2790 (1987), ISSN 1079-7114.

[26] P. W. Anderson, Mater. Res. Bull. **8**, 153 (1973), ISSN 0025-5408.

[27] C. Gros, R. Joynt, and T. M. Rice, Z. Phys. B: Condens. Matter **68**, 425 (1987), ISSN 1431-584X.

[28] D. J. Scalapino, E. Loh, and J. E. Hirsch, Phys. Rev. B **35**, 6694 (1987), ISSN 2469-9969.

[29] J. E. Hirsch and H. Q. Lin, Phys. Rev. B **37**, 5070 (1988), ISSN 2469-9969.

[30] H. Q. Lin, J. E. Hirsch, and D. J. Scalapino, Phys. Rev. B **37**, 7359 (1988), ISSN 2469-9969.

[31] N. E. Bickers, D. J. Scalapino, and S. R. White, Phys. Rev. Lett. **62**, 961 (1989), ISSN 1079-7114.

[32] P. van Dongen and D. Vollhardt, Phys. Rev. B **40**, 7252 (1989), ISSN 2469-9969.

[33] E. H. Lieb, Phys. Rev. Lett. **62**, 1201 (1989), ISSN 1079-7114.

[34] W. Metzner and D. Vollhardt, Phys. Rev. Lett. **62**, 324 (1989), ISSN 1079-7114.

[35] A. Georges and G. Kotliar, Phys. Rev. B **45**, 6479 (1992), ISSN 2469-9969.

[36] S. R. White, D. J. Scalapino, R. L. Sugar, E. Y. Loh, J. E. Gubernatis, and R. T. Scalettar, Phys. Rev. B **40**, 506 (1989), ISSN 2469-9969.

[37] S. R. White, Phys. Rev. Lett. **69**, 2863 (1992), ISSN 1079-7114.

[38] S. R. White, Phys. Rev. B **48**, 10345 (1993), ISSN 2469-9969.

[39] S. Östlund and S. Rommer, Phys. Rev. Lett. **75**, 3537 (1995), ISSN 1079-7114.

[40] I. Affleck, T. Kennedy, E. H. Lieb, and H. Tasaki, SpringerLink pp. 253–304 (1988).

[41] M. Fannes, B. Nachtergaele, and R. F. Werner, Commun. Math. Phys. **144**, 443 (1992), ISSN 1432-0916.

[42] S. Liang and H. Pang, Phys. Rev. B **49**, 9214 (1994), ISSN 2469-9969.

[43] Y.-Y. Shi, L.-M. Duan, and G. Vidal, Phys. Rev. A **74**, 022320 (2006), ISSN 2469-9934.

[44] N. Nakatani and G. K.-L. Chan, J. Chem. Phys. **138**, 134113 (2013), ISSN 0021-9606.

[45] V. Murg, F. Verstraete, R. Schneider, P. R. Nagy, and Ö. Legeza, J. Chem. Theory Comput. **11**, 1027 (2015), ISSN 1549-9618.

[46] V. Murg, F. Verstraete, Ö. Legeza, and R. M. Noack, Phys. Rev. B **82**, 205105 (2010), ISSN 2469-9969.

[47] K. Gunst, F. Verstraete, S. Wouters, Ö. Legeza, and D. Van Neck, J. Chem. Theory Comput. **14**, 2026 (2018), ISSN 1549-9618.

[48] L. P. Kadanoff, Physics Physique Fizika **2**, 263 (1966).

[49] P. Corboz and G. Vidal, Phys. Rev. B **80**, 165129 (2009), ISSN 2469-9969.

[50] P. Corboz, G. Evenbly, F. Verstraete, and G. Vidal, Phys. Rev. A **81**, 010303(R) (2010), ISSN 2469-9934.

[51] F. Verstraete and J. I. Cirac, arXiv (2004), `cond-mat/0407066`, URL `https://arxiv.org/abs/cond-mat/0407066`.

[52] F. Verstraete and J. I. Cirac, Phys. Rev. A **70**, 060302(R) (2004), ISSN 2469-9934.

[53] D. Perez-Garcia, F. Verstraete, J. I. Cirac, and M. M. Wolf, arXiv (2007), 0707.2260, URL https://arxiv.org/abs/0707.2260.

[54] N. Schuch, I. Cirac, and D. Pérez-García, Ann. Phys. **325**, 2153 (2010), ISSN 0003-4916.

[55] F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac, Phys. Rev. Lett. **96**, 220601 (2006), ISSN 1079-7114.

[56] N. Schuch, D. Pérez-García, and I. Cirac, Phys. Rev. B **84**, 165139 (2011), ISSN 2469-9969.

[57] V. Murg, F. Verstraete, and J. I. Cirac, Phys. Rev. A **75**, 033605 (2007), ISSN 2469-9934.

[58] V. Murg, F. Verstraete, and J. I. Cirac, Phys. Rev. B **79**, 195119 (2009), ISSN 2469-9969.

[59] J. Jordan, R. Orús, G. Vidal, F. Verstraete, and J. I. Cirac, Phys. Rev. Lett. **101**, 250602 (2008), ISSN 1079-7114.

[60] P. Corboz, R. Orús, B. Bauer, and G. Vidal, Phys. Rev. B **81**, 165104 (2010), ISSN 2469-9969.

[61] P. Corboz, S. R. White, G. Vidal, and M. Troyer, Phys. Rev. B **84**, 041108(R) (2011), ISSN 2469-9969.

[62] P. Corboz, J. Jordan, and G. Vidal, Phys. Rev. B **82**, 245119 (2010), ISSN 2469-9969.

[63] P. Corboz, T. M. Rice, and M. Troyer, Phys. Rev. Lett. **113**, 046402 (2014), ISSN 1079-7114.

[64] B.-X. Zheng, C.-M. Chung, P. Corboz, G. Ehlers, M.-P. Qin, R. M. Noack, H. Shi, S. R. White, S. Zhang, and G. K.-L. Chan, Science **358**, 1155 (2017), ISSN 0036-8075.

[65] C. Lanczos, J. Res. Matl. Bur. Stand. **45**, 225 (1950).

[66] E. R. Davidson, J. Comput. Phys. **17**, 87 (1975).

[67] R. M. Noack and S. R. Manmana, AIP Conference Proceedings **789**, 93 (2005), ISSN 0094-243X.

[68] E. R. Davidson, Comput. Phys. Commun. **53**, 49 (1989), ISSN 0010-4655.

[69] R. B. Morgan, J. Comput. Phys. **89**, 241 (1990), ISSN 0021-9991.

[70] A. Weichselbaum, Ann. Phys. **327**, 2972 (2012), ISSN 0003-4916.

[71] F. H. L. Essler, H. Frahm, F. Göhmann, A. Klümper, and V. E. Korepin, *The One-Dimensional Hubbard Model* (Cambridge University Press, Cambridge, England, UK, 2005), ISBN 978-0-52180262-8.

[72] A. Messiah, *Quantum Mechanics, Volume II* (North-Holland Publishing Compan, 1966).

[73] F. D. M. Haldane, Phys. Lett. A **93**, 464 (1983), ISSN 0375-9601.

[74] F. D. M. Haldane, Phys. Rev. Lett. **50**, 1153 (1983), ISSN 1079-7114.

[75] F. D. M. Haldane, J. Appl. Phys. **57**, 3359 (1985), ISSN 0021-8979.

[76] F. Verstraete, V. Murg, and J. I. Cirac, Adv. Phys. **57**, 143 (2008), ISSN 0001-8732.

[77] J. Preskill, *Quantum Information and Computation* (2019), [Online; accessed 26. Mar. 2020], URL `http://www.theory.caltech.edu/people/preskill/ph229/index.html`.

[78] C. E. Shannon, Bell System Technical Journal **27**, 379 (1948), ISSN 0005-8580.

[79] J. Eisert, M. Cramer, and M. B. Plenio, Rev. Mod. Phys. **82**, 277 (2010), ISSN 1539-0756.

[80] K. Audenaert, J. Eisert, M. B. Plenio, and R. F. Werner, Phys. Rev. A **66**, 042327 (2002), ISSN 2469-9934.

[81] M. B. Plenio, Phys. Rev. Lett. **95**, 090503 (2005), ISSN 1079-7114.

[82] B.-Q. Jin and V. E. Korepin, J. Stat. Phys. **116**, 79 (2004), ISSN 1572-9613.

[83] A. R. Its, B.-Q. Jin, and V. E. Korepin, J. Phys. A: Math. Gen. **38**, 2975 (2005), ISSN 0305-4470.

[84] J. P. Keating and F. Mezzadri, Phys. Rev. Lett. **94**, 050501 (2005), ISSN 1079-7114.

[85] M. B. Hastings, J. Stat. Mech.: Theory Exp. **2007**, P08024 (2007), ISSN 1742-5468.

[86] E. H. Lieb and D. W. Robinson, SpringerLink pp. 425–431 (1972).

[87] G. Refael and J. E. Moore, Phys. Rev. Lett. **93**, 260602 (2004), ISSN 1079-7114.

[88] G. Refael and J. E. Moore, Phys. Rev. B **76**, 024419 (2007), ISSN 2469-9969.

[89] M. Cramer, J. Eisert, M. B. Plenio, and J. Dreißig, Phys. Rev. A **73**, 012309 (2006), ISSN 2469-9934.

[90] M. Cramer and J. Eisert, New J. Phys. **8**, 71 (2006), ISSN 1367-2630.

[91] M. M. Wolf, Phys. Rev. Lett. **96**, 010404 (2006), ISSN 1079-7114.

[92] F. F. Assaad, T. C. Lang, and F. Parisen Toldin, Phys. Rev. B **89**, 125121 (2014), ISSN 2469-9969.

[93] F. F. Assaad, Phys. Rev. B **91**, 125146 (2015), ISSN 2469-9969.

[94] F. Parisen Toldin and F. F. Assaad, Phys. Rev. Lett. **121**, 200602 (2018), ISSN 1079-7114.

[95] F. P. Toldin and F. F. Assaad, IOP Publishing **1163**, 012056 (2019), ISSN 1742-6588.

[96] B. Bauer, P. Corboz, R. Orús, and M. Troyer, Phys. Rev. B **83**, 125106 (2011), ISSN 2469-9969.

[97] S. Singh, R. N. C. Pfeifer, and G. Vidal, Phys. Rev. B **83**, 115125 (2011), ISSN 2469-9969.

[98] S. Singh, R. N. C. Pfeifer, and G. Vidal, Phys. Rev. A **82**, 050301(R) (2010), ISSN 2469-9934.

[99] S. Singh and G. Vidal, Phys. Rev. B **86**, 195114 (2012), ISSN 2469-9969.

[100] M. Fishman, E. M. Stoudenmire, and S. R. White, *ITensor - Intelligent Tensor Library* (2019), [Online; accessed 27. Mar. 2020], URL `http://itensor.org/about.html`.

[101] M. Lubasch, J. I. Cirac, and M.-C. Bañuls, Phys. Rev. B **90**, 064425 (2014), ISSN 2469-9969.

[102] G. Vidal, Phys. Rev. Lett. **91**, 147902 (2003), ISSN 1079-7114.

[103] F. Verstraete and J. I. Cirac, Phys. Rev. B **73**, 094423 (2006), ISSN 2469-9969.

[104] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, arXiv (2006), `quant-ph/0608197`, URL `https://arxiv.org/abs/quant-ph/0608197`.

[105] U. Schollwöck, Ann. Phys. **326**, 96 (2011), ISSN 0003-4916.

[106] G. M. Crosswhite, A. C. Doherty, and G. Vidal, Phys. Rev. B **78**, 035116 (2008), ISSN 2469-9969.

[107] G. M. Crosswhite and D. Bacon, Phys. Rev. A **78**, 012356 (2008), ISSN 2469-9934.

[108] R. M. Noack, S. R. White, and D. J. Scalapino, Phys. Rev. Lett. **73**, 882 (1994), ISSN 1079-7114.

[109] S. R. White, R. M. Noack, and D. J. Scalapino, Phys. Rev. Lett. **73**, 886 (1994), ISSN 1079-7114.

[110] R. M. Noack, S. R. White, and D. J. Scalapino, EPL **30**, 163 (1995), ISSN 0295-5075.

[111] R. M. Noack, S. R. White, and D. J. Scalapino, Physica C **270**, 281 (1996), ISSN 0921-4534.

[112] S. R. White and D. J. Scalapino, Phys. Rev. Lett. **80**, 1272 (1998), ISSN 1079-7114.

[113] R. M. Noack, N. Bulut, D. J. Scalapino, and M. G. Zacher, Phys. Rev. B **56**, 7162 (1997), ISSN 2469-9969.

[114] S. R. White and D. J. Scalapino, Phys. Rev. Lett. **81**, 3227 (1998), ISSN 1079-7114.

[115] S. R. White and D. J. Scalapino, Phys. Rev. B **60**, R753(R) (1999), ISSN 2469-9969.

[116] S. Yan, D. A. Huse, and S. R. White, Science **332**, 1173 (2011), ISSN 0036-8075.

[117] E. M. Stoudenmire and S. R. White, Annu. Rev. Condens. Matter Phys. **3**, 111 (2012), ISSN 1947-5454.

[118] S. Depenbrock, I. P. McCulloch, and U. Schollwöck, Phys. Rev. Lett. **109**, 067201 (2012), ISSN 1079-7114.

[119] Simons Collaboration on the Many-Electron Problem, J. P. F. LeBlanc, A. E. Antipov, F. Becca, I. W. Bulik, G. K.-L. Chan, C.-M. Chung, Y. Deng, M. Ferrero, T. M. Henderson, et al., Phys. Rev. X **5**, 041041 (2015), ISSN 2160-3308.

[120] G. Ehlers, J. Sólyom, Ö. Legeza, and R. M. Noack, Phys. Rev. B **92**, 235116 (2015), ISSN 2469-9969.

[121] G. Ehlers, S. R. White, and R. M. Noack, Phys. Rev. B **95**, 125125 (2017), ISSN 2469-9969.

[122] G. Ehlers, B. Lenz, S. R. Manmana, and R. M. Noack, Phys. Rev. B **97**, 035118 (2018), ISSN 2469-9969.

[123] F. Fröwis, V. Nebendahl, and W. Dür, Phys. Rev. A **81**, 062337 (2010), ISSN 2469-9934.

[124] F. Verstraete, J. J. García-Ripoll, and J. I. Cirac, Phys. Rev. Lett. **93**, 207204 (2004), ISSN 1079-7114.

[125] I. Pižorn, L. Wang, and F. Verstraete, Phys. Rev. A **83**, 052321 (2011), ISSN 2469-9934.

[126] M. Lubasch, J. I. Cirac, and M.-C. Bañuls, New J. Phys. **16**, 033014 (2014), ISSN 1367-2630.

[127] H. C. Jiang, Z. Y. Weng, and T. Xiang, Phys. Rev. Lett. **101**, 090603 (2008), ISSN 1079-7114.

[128] P. Corboz, Phys. Rev. B **94**, 035133 (2016), ISSN 2469-9969.

[129] H. N. Phien, J. A. Bengua, H. D. Tuan, P. Corboz, and R. Orús, Phys. Rev. B **92**, 035142 (2015), ISSN 2469-9969.

[130] G. Evenbly, Phys. Rev. B **98**, 085155 (2018), ISSN 2469-9969.

[131] P. Corboz, Phys. Rev. B **93**, 045116 (2016), ISSN 2469-9969.

[132] *Stripe order in the underdoped region of the two-dimensional Hubbard model, Supplementary Materials* (2021), [Online; accessed 31. Jan. 2021], URL `https://science.sciencemag.org/content/suppl/2017/12/01/358.6367.1155.DC1`.

[133] V. J. Emery, Phys. Rev. Lett. **58**, 2794 (1987), ISSN 1079-7114.

Die Seite 97 (Lebenslauf/Curriculum Vitae) enthält persönliche Daten. Sie ist deshalb nicht Bestandteil der Online-Veröffentlichung.

Acknowledgements

I would like to thank Reinhard Noack for the supervision of my thesis, Georg Ehlers for introducing me into the details of the DMRG, C++, and the vocabulary of North Germans, Florian Gebhard for making sure that I wouldn't have to fulfill my teaching duties in laboratory courses, Alexander Orlov for a lot of memorable nights in the clubs of Marburg, Jan Thiede for the eating orgies during lunch breaks, Thorben Linneweber and David Geisel for exciting chess matches, the Hessian taxpayer for paying my rent, and my parents for patiently waiting for the end of this trial.